NUMERICAL ANALYSIS OF INTRAGRAIN PRECIPITATE INFLUENCE ON THE LINEAR SCATTERING BEHAVIOR

A Thesis
Presented to
The Academic Faculty

by

Daniel Gruen

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in Engineering Science and Mechanics in the
School of Civil and Environmental Engineering

Georgia Institute of Technology
December 2016

NUMERICAL ANALYSIS OF INTRAGRAIN PRECIPITATE INFLUENCE ON THE LINEAR SCATTERING BEHAVIOR

Approved by:

Professor Laurence J. Jacobs, Advisor
School of Civil and Environmental Engineering
Georgia Institute of Technology

Professor Laurent Capolungo
Materials Science and Technology Division
Los Alamos National Laboratory

Dr. Jin-Yeon Kim
School of Civil and Environmental Engineering
Georgia Institute of Technology

Date Approved: August 26, 2016

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest appreciation to my adviser Professor Laurence J. Jacobs for the support and the highly motivating atmosphere he provided me during my time at Georgia Tech. He made it possible for me to come to Georgia Tech, write a thesis in his lab, and earn one of the best engineering degrees in the United States. Thank you Larry for the time you spent with us talking and caring about academic as well as private issues, the coffee and ice cream scholarships, and the unforgettable year we could spend in Atlanta.

Thank you to Professor Laurent Capolungo for your guidance throughout my thesis, spending hours in your office working on differential equations, and always keeping me motivated. I learned a lot about high performance computing and numerical methods from you, and I would not be the diligent researcher I am today without your guidance.

Thank you to Dr. Jin-Yeon Kim for your invaluable advice and help with complex wave propagation problems, enlightening discussions, and with anything and everything else around the lab. I really enjoyed working with you in the lab.

Moreover, thanks go to my lab mates Kathy Scott, David Torello, Gun Kim, Nick Selby, Marc Forstenhaeusler and Christoph Doerr who made the lab an enjoyable place to work, meet and talk. I would in particular like to thank Gun Kim for his insightful conversations and encouraging in difficult times.

Special thanks go to Professor Lothar Gaul and Christian Ehrlich from the Institute of Applied and Experimental Mechanics at the University of Stuttgart for choosing me as a candidate for the ISAP Program, which is financially supported by

the German Academic Exchange Service DAAD. The DAAD is gratefully acknowledged for supporting me with its scholarship.

Finally, I would like to express my deepest gratitude to my family for supporting and caring about me, especially to my parents who enabled me to go to college and get admitted to spend a memorable year at Georgia Tech.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS OR ABBREVIATIONS

∂	partial derivative operator
δ	variational operator
$\Im$	Fourier transformation operator
$*$	convolution operator
α	acoustic attenuation coefficient
β	acoustic nonlinearity parameter
δ_{ij}	Kronecker delta
ϵ_{ij}	strain tensor
θ	azimuth in cylindrical coordinate system
λ	wavelength or first Lamé constant
μ	second Lamé constant
ν	Poisson's ratio
π	mathematical constant
ρ	mass density
σ_{ij}	stress tensor
τ_{ij}	shear stress tensor
ϕ	number density or interpolation function
ω	angular frequency
Δ	Laplace operator
Ω	domain or region
a	defect size
c	velocity
c_p	pressure wave speed

c_s	shear wave speed
d	diameter
d_{ij}	Euclidean distance matrix
e	exponential function
f	frequency
f_i	body force vector
${}^{n}f^{j}$	one-point statistics
${}^{np}f_{t}^{j}$	two-point statistics
h	local state or element size
i	imaginary unit
k	wave number or bulk modulus
ka	normalized frequency
l	edge length of squared domain
m^j	probability density function to find distinct local state
n_i	normal vector
r	radius or radial axis in cylindrical coordinate system
rr_{ij}	range ratio of i^{th} and j^{th} principal components
t	time or random vector
t_i	traction vector
u	displacement along first coordinate axis
u_i	displacement vector
v	displacement along second coordinate axis or test function
w	displacement along third coordinate axis
x	first coordinate axis
y	second coordinate axis
z	third coordinate axis
x_i	i^{th} axis or direction in coordinate system

A	amplitude
C_{ijkl}	higher order stiffness tensor
D_{ijkl}	higher order stiffness tensor
E	Young's modulus of elasticity
F_i	force vector
H_n^i	Hankel function
N	number of precipitates
S	surface or uniform grid size
U	internal work or approximated displacement
V	volume
W	external work

max	maximum
min	minimum
nm	nanometer
var	variance
CFL	CourantFriedrichsLewy
CG	conjugate gradient
Cu	element symbol of copper
Fe	element symbol of iron
FEM	finite element method
KDE	kernel density estimation
MS	microstructure
PC	principal component
PCA	principal component analysis

SUMMARY

Nondestructive evaluation using ultrasonic waves is commonly used to experimentally probe for the presence of defects (i.e. dislocations, precipitates, cracks) in complex metallic microstructures. Such defects and abnormalities are evidenced by monitoring the acoustic attenuation coefficient α. However, from a mathematical standpoint, the correlation between the microstructural behavior and the measured acoustic attenuation behavior is not yet explicit. The present work aims to assess the existence of statistical correlations between microstructural defects and acoustic attenuation. The effects of defect geometry, density, and geometrical arrangements (i.e. relative position) on acoustic attenuation are studied. To do so, the acoustic response of Fe-Cu single crystals containing 1 % Cu precipitates with radii on the order of 2 nm is simulated by means of finite element analysis. Several thousand initial microstructures with random arrangement of precipitates are virtually tested using statistical methods, such as principal component analysis. Therefore, it is expected that a causal link can be made between the acoustic attenuation coefficient and the precipitates-induced microstructural behavior via the proposed numerical analysis.

CHAPTER I

INTRODUCTION

1.1 Motivation and Objective

Nondestructive evaluation techniques using ultrasonic waves are commonly used to experimentally probe for the presence of defects, such as cracks, precipitates, and dislocations in complex metallic microstructures by monitoring the attenuation coefficient and acoustic nonlinearity parameter. Many components in technical applications are critical for a safe operation and therefore a continuous evaluation throughout the lifetime of the changes in the material is necessary.

Since the measurement of the material acoustic properties, namely, the attenuation coefficient α do not provide the information on the exact defect type present, numerical analyses need to be performed in order to assess the eventual influence of each defect type. The defect type of main interest in this work are precipitates or more specifically intragrain precipitates, in other words precipitates inside of a grain. All other defect types are assumed to be nonexistent.

The first unknown in the case of the influence of intragrain precipitates on the acoustic properties is how the arrangement of the precipitates affect the attenuation coefficient α and the acoustic nonlinearity parameter β. Assuming a nonlinear stress-strain relation as in (1), the exact form of the resulting wave equation would be (2).

$$\sigma_{ij} = C_{ijkl}\epsilon_{kl} + \frac{1}{2}C_{ijklmn}\epsilon_{kl}\epsilon_{mn} + ... \tag{1}$$

$$\rho\frac{\partial^2 u_i}{\partial t^2} = \frac{\partial \sigma_{ij}}{\partial x_j} = C_{ijkl}\epsilon_{kl,j} + C_{ijkl,j}\epsilon_{kl} + \frac{1}{2}C_{ijklmn}\left(\epsilon_{kl,j}\epsilon_{mn}\right) + \frac{1}{2}C_{ijklmn,j}\left(\epsilon_{kl}\epsilon_{mn}\right) \tag{2}$$

$$\rho\frac{\partial^2 u_i}{\partial t^2} = \frac{\partial \sigma_{ij}}{\partial x_j} = \left(C_{ijkl} + C_{ijklmn}\epsilon_{mn}\right)\epsilon_{kl,j} \tag{3}$$

However, most derivations usually neglect the influence of spatially changing material properties although precipitates clearly constitute a heterogeneous material. The commonly used form of the wave equation is therefore equation (3). The question that remains open whether $C_{ijkl,j}$ and $C_{ijklmn,j}$ have a major impact on the accuracy of the results. Another aspect is if or how sensitive acoustic parameters are to other material properties, such as volume fraction, size, and mechanical properties of the precipitates.

Previous works by Cantrell and Zhang [3], Matlack [13], Thiele [20] as well as Herrmann [8] mainly focus on the influence of the comparatively large defects dislocations and cracks. Even though almost every defect type originates from a certain precipitate behavior, the precipitates themselves are usually not considered as a main impact on acoustic properties.

To fill this gap, the first objective of this work is to numerically show the existence of a correlation between precipitates as a microstructural defect type and the acoustic attenuation property α by applying a statistical analysis. To do so, a large number of initially same microstructures with random arrangements of precipitates are virtually generated and the attenuation coefficient is determined for each structure by solving the wave equation numerically.

After showing the existence of a correlation, the second part of the objective is to quantify how the spatial distribution of precipitates affects the attenuation coefficient α. Thereto, the microstructural information on the one side and the acoustic properties on the other are linked together and examined with the principal component analysis.

The material studied in this research is a iron - 1% copper binary alloy, a well-suited material to investigate precipitation caused by radiation damage. A validation of the current work is possible with Scott's study [19].

1.2 Structure of Thesis

An introduction into the fundamentals of wave propagation will be given in Chapter 2 leading to the derivation of the stress-strain relation and basic characteristics of linear wave propagation. Moreover, a brief explanation of wave scattering and attenuation is provided. Chapter 3 discusses the statistical methods principal component analysis and two-point statistics that are utilized to evaluate the correlation between microstructural and acoustic properties. The main part of this work, the numerical analysis, including a short section about finite element methods, actual material properties, and the structure of this specific analysis is presented in the Chapter 4. Chapter 5 presents all the results and findings of the present study. Finally, Chapter 6 draws conclusions and gives an outlook for future work.

CHAPTER II

WAVE PROPAGATION IN ELASTIC SOLIDS

This chapter introduces and explains the basics of wave propagation in elastic solids. Starting off with the linear stress strain relation, wave propagation and its weak form that is necessary for the finite element solver that is used in this work and proceeding with the backgrounds of Transmission, Reflection, Scattering and Attenuation, this chapter illuminates essential parts of the great area of wave propagation and sets the main foundation for this thesis.

2.1 Elastic Material and Linear Stress Strain Relation

To have a common basis and distinct use of language, a clear definition of the investigated case and all related terms is essential. Therefore, the definition of an elastic material is regarded in the first instance. An elastic material is a material that has a one-to-one correspondence between stress and strain. An elastic material follows the same stress-strain path during loading and unloading and the strain density function U_0 exists which can be expressed in terms of the state of current strain only, independent of the strain history or strain path. Note that the elastic material does not necessarily mean that the stress-strain relation is linear. If the stress-strain path is different during loading and unloading, then the material is no longer elastic even if the path is linear during loading and unloading.

For conservative materials, that is materials with no energy dissipation, the external work done on the material has to be equal to the total change in the strain energy of the material. The latter relation can be expressed as

$$\delta W = \delta U \tag{4}$$

where δW and δU are the variations of the external work done on the material body and the variation of internal energy of the material body. The variation of the external work can be rewritten in terms of the applied body force f_i and the surface traction t_i (i indicates summation over all three directions)

$$\delta W = \int_V f_i \delta u_i dV + \int_S t_i \delta u_i dS. \tag{5}$$

Also, the variation of the internal energy corresponds to the volume integral of the strain energy density variation δU_0

$$\delta U = \int_V \delta U_0 dV. \tag{6}$$

Using equations (5), (6), and the definition of the surface traction t_i

$$t_i = \sigma_{ij} n_j, \tag{7}$$

the energy balance equation (4) can be reformulated as

$$\begin{aligned} \int_V \delta U_0 dV &= \int_V f_i \delta u_i dV + \int_S t_i \delta u_i dS = \int_V f_i \delta u_i dV + \int_S \sigma_{ij} n_j \delta u_i dS \\ &= \int_V f_i \delta u_i dV + \int_S \left(\sigma_{ij} \delta u_i\right) n_j dS. \end{aligned} \tag{8}$$

Applying the divergence theorem on the surface integral of the right hand side, one gets

$$\begin{aligned} \int_V \delta U_0 dV &= \int_V f_i \delta u_i dV + \int_V \left(\sigma_{ij} \delta u_i\right)_{,j} dV \\ &= \int_V f_i \delta u_i dV + \int_V \left(\sigma_{ij,j}\, \delta u_i + \sigma_{ij} \delta u_{i,j}\right) dV \\ &= \int_V \left(f_i \delta u_i + \sigma_{ij,j}\, \delta u_i + \sigma_{ij} \delta u_{i,j}\right) dV \\ &= \int_V \left(\left(f_i + \sigma_{ij,j}\right) \delta u_i + \sigma_{ij} \delta u_{i,j}\right) dV. \end{aligned} \tag{9}$$

With the force equilibrium for an elemental volume

$$\sigma_{ij,j} + f_i = 0 \tag{10}$$

equation (9) is simplified to

$$\begin{aligned}\int_V \delta U_0 dV &= \int_V \left(\sigma_{ij}\delta u_{i,j}\right) dV \\ &= \int_V \frac{1}{2}\left(\delta u_{i,j} + \delta u_{j,i}\right) dV \\ &= \int_V \sigma_{ij}\delta\epsilon_{ij} dV. \end{aligned} \tag{11}$$

Since equation (11) is valid for any volume V, both integrands have to be equal to each other

$$\delta U_0 = \sigma_{ij}\delta\epsilon_{ij} \tag{12}$$

and therefore the strain energy density U_0 is only a function of the strain ϵ_{ij}. Including the given definition of elastic materials, equation (12) concludes with

$$\sigma_{ij} = \frac{\partial U_0}{\partial \epsilon_{ij}}. \tag{13}$$

The strain energy density function $U_0\left(\epsilon_{ij}\right)$ is usually assumed to be a complete second-degree polynomial of the form

$$U_0 = D_0 + D_{kl}\epsilon_{kl} + D_{klmn}\epsilon_{kl}\epsilon_{mn} \tag{14}$$

and thus

$$\begin{aligned}\sigma_{ij} &= \frac{\partial U_0}{\partial \epsilon_{ij}} \\ &= D_{kl}\delta_{ik}\delta_{jl} + D_{klmn}\left(\delta_{ik}\delta_{jl}\epsilon_{mn} + \epsilon_{kl}\delta_{im}\delta_{jn}\right) \\ &= D_{ij} + D_{ijmn}\epsilon_{mn} + D_{klij}\epsilon_{kl} \\ &= D_{ij} + \left(D_{ijkl} + D_{klij}\right)\epsilon_{kl}. \end{aligned} \tag{15}$$

With $\left(D_{ijkl} + D_{klij}\right) = C_{ijkl}$ (C_{ijkl} is the fourth order stiffness tensor) and $D_{ij} = 0$ (no residual stress assumption) the linear constitutive relation

$$\sigma_{ij} = C_{ijkl}\epsilon_{kl} \tag{16}$$

is obtained.

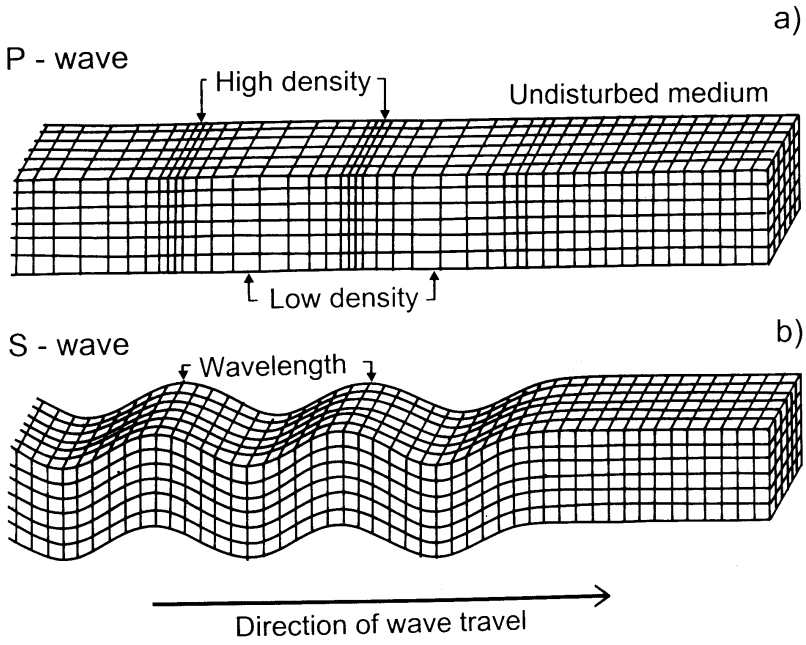


Figure 1: Wave types

In the same manner, a nonlinear, quadratic material will have the constitutive relation

$$\sigma_{ij} = C_{ijkl}\epsilon_{kl} + C_{ijklmn}\epsilon_{kl}\epsilon_{mn}. \tag{17}$$

2.2 Equation of Motion

In this section the equation of motion of an isotropic elastic medium will be derived in terms of particle displacements. It will be shown that these equations of motion correspond to two types of waves which can propagate through an elastic solid. These two types of wave are called dilatational or pressure wave (p-wave) and distortional or shear wave (s-wave). The particle motion in a plane dilatational wave is along the direction of propagation, whereas in a plane distortional wave the particle motion is perpendicular to the direction of propagation.

If the solid is unbounded these are the only existing types of wave. When the solid has a free surface or where surface boundaries exist between two solids Rayleigh

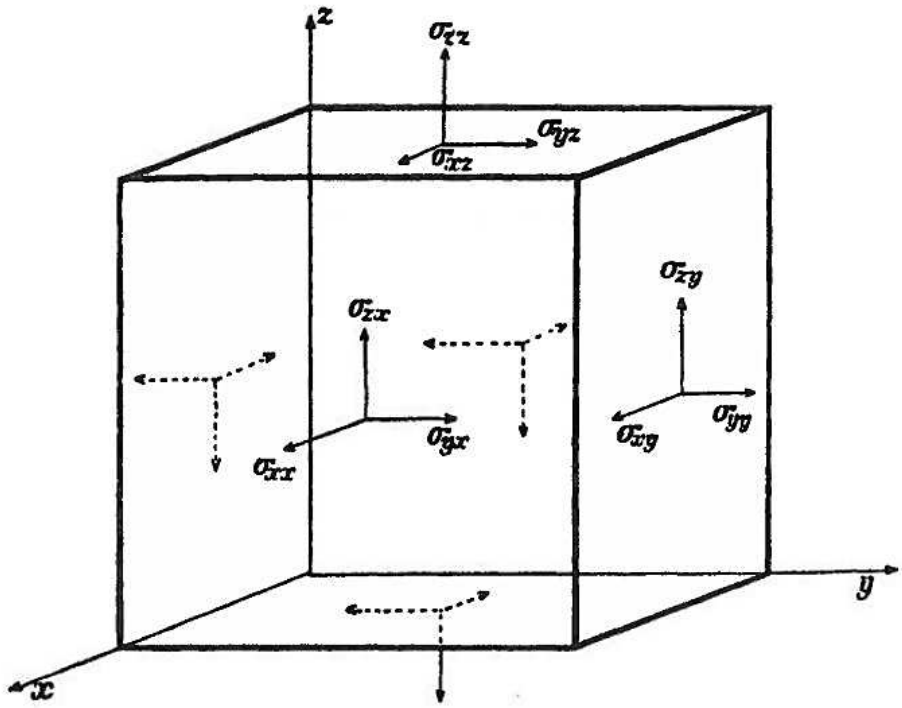


Figure 2: Stress components acting on an infinitesimal rectangular cube

surface waves may be present. Since Rayleigh wave measurements require the accessibility of only one surface this method is highly qualified for on-site scrutinies of structures. The works of Thiele [20] and Doerr [6] focus on Rayleigh wave measurements to determine the relative acoustic nonlinearity parameter β.

Components of Stress and Strain

The stress on a surface element in a solid does usually not act solely normally on that surface but has components both normal and tangential to that plane. If the axes of a three-dimensional, orthogonal coordinate system are referred to as x-, y-, and z-axis and stresses are considered in every direction on planes perpendicular to each axis, there will be nine stress components for the volume element. These components are denoted by σ_{xx}, σ_{yy}, σ_{zz}, σ_{xy}, σ_{xz}, σ_{yx}, σ_{yz}, σ_{zx}, and σ_{zy}. The first letter in the suffixes indicates the direction of the stress and the second letter the plane in which it is acting.

Considering an infinitesimal, equilateral, and rectangular cube with its faces normal to the axes (see Fig. 2) and taking the sum of all forces and moments, it may be seen that for equilibrium

$$\sigma_{xy} = \sigma_{yx} \tag{18}$$

$$\sigma_{xz} = \sigma_{zx} \tag{19}$$

$$\sigma_{yz} = \sigma_{zy}, \tag{20}$$

so that only six independent components remain. With that, the general stress tensor

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix} = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix} = \begin{bmatrix} \sigma_{x} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_{y} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_{z} \end{bmatrix} \tag{21}$$

can be rewritten as the simplified engineering stress vector $\boldsymbol{\sigma_{eng}}$. This representation of the stress tensor is often used in engineering calculations or for material property determination and is only valid for elastic and isotropic materials.

$$\boldsymbol{\sigma_{eng}} = \begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{xy} \\ \sigma_{xz} \\ \sigma_{yz} \end{Bmatrix} = \begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{Bmatrix} = \begin{Bmatrix} \sigma_{x} \\ \sigma_{y} \\ \sigma_{z} \\ \tau_{xy} \\ \tau_{xz} \\ \tau_{yz} \end{Bmatrix} \tag{22}$$

The displacement of any point in the body may be represented as a superposition of displacements u,v, and w parallel to the x, y, and z axes. In order to find the strain at a point P in the body, the displacement of P has to be determined relative to its adjacent points. Considering a point very close to P, which in the undisplaced position had the coordinates $(x+\delta x)$, $(y+\delta y)$, $(z+\delta z)$, and displacement components $(u+\delta u)$, $(v+\delta v)$, $(w+\delta w)$, following relations result for sufficiently small δx, δy,

and δz:

$$\delta u = \frac{\partial u}{\partial x}\delta x + \frac{\partial u}{\partial y}\delta y + \frac{\partial u}{\partial z}\delta z, \tag{23}$$

$$\delta v = \frac{\partial v}{\partial x}\delta x + \frac{\partial v}{\partial y}\delta y + \frac{\partial v}{\partial z}\delta z, \tag{24}$$

$$\delta w = \frac{\partial w}{\partial x}\delta x + \frac{\partial w}{\partial y}\delta y + \frac{\partial w}{\partial z}\delta z. \tag{25}$$

Thus, if the nine quantities from the three equations above are known for a distinct point P, the displacements of all surrounding points may be found. Usually, these nine quantities are regrouped and denoted as the strain components ϵ_{xx}, ϵ_{yy}, ϵ_{zz}, ϵ_{yz}, ϵ_{zx}, and ϵ_{xy}.

$$\epsilon_{xx} = \frac{\partial u}{\partial x} \tag{26}$$

$$\epsilon_{yy} = \frac{\partial v}{\partial y} \tag{27}$$

$$\epsilon_{zz} = \frac{\partial w}{\partial z} \tag{28}$$

$$\epsilon_{yz} = \frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \tag{29}$$

$$\epsilon_{zx} = \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \tag{30}$$

$$\epsilon_{xy} = \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \tag{31}$$

The first three strain components ϵ_{xx}, ϵ_{yy}, and ϵ_{zz} correspond to the fractional expansions and contractions of infinitesimal line elements passing through the point P and parallel to x, y, and z respectively. The last three components ϵ_{yz}, ϵ_{zx}, and ϵ_{xy} represent the shear strain amount in the planes indicated by their suffixes.

Generalized form of Hooke's law

For most solids it is found experimentally that the measured strains are proportional to the applied load, as long as the load does not exceed a certain value which is known as the elastic limit of a material. Therefore, each of the six components of stress is at any point a linear function of the six components of strain (Kolsky [11]). Whilst the

law in this form is incapable of direct experimental proof, it summarizes the results of different types of experimental loading. Wherever the mathematical consequences can be tested, the formulated law is found to be true within the elastic range of a material.

The generalized form of Hooke's law is finally stated as

$$\sigma_{xx} = C_{11}\epsilon_{xx} + C_{12}\epsilon_{yy} + C_{13}\epsilon_{zz} + C_{14}\epsilon_{yz} + C_{15}\epsilon_{zx} + C_{16}\epsilon_{xy} \tag{32}$$

$$\sigma_{yy} = C_{21}\epsilon_{xx} + C_{22}\epsilon_{yy} + C_{23}\epsilon_{zz} + C_{24}\epsilon_{yz} + C_{25}\epsilon_{zx} + C_{26}\epsilon_{xy} \tag{33}$$

$$\sigma_{zz} = C_{31}\epsilon_{xx} + C_{32}\epsilon_{yy} + C_{33}\epsilon_{zz} + C_{34}\epsilon_{yz} + C_{35}\epsilon_{zx} + C_{36}\epsilon_{xy} \tag{34}$$

$$\sigma_{yz} = C_{41}\epsilon_{xx} + C_{42}\epsilon_{yy} + C_{43}\epsilon_{zz} + C_{44}\epsilon_{yz} + C_{45}\epsilon_{zx} + C_{46}\epsilon_{xy} \tag{35}$$

$$\sigma_{zx} = C_{51}\epsilon_{xx} + C_{52}\epsilon_{yy} + C_{53}\epsilon_{zz} + C_{54}\epsilon_{yz} + C_{55}\epsilon_{zx} + C_{56}\epsilon_{xy} \tag{36}$$

$$\sigma_{xy} = C_{61}\epsilon_{xx} + C_{62}\epsilon_{yy} + C_{63}\epsilon_{zz} + C_{64}\epsilon_{yz} + C_{65}\epsilon_{zx} + C_{66}\epsilon_{xy} \tag{37}$$

where the coefficients C_{ij} are the elsatic constants of the material.

The number of independent coefficients can be reduced from 36 to 21 by showing that the condition for the elastic energy to be an injective function of the strain is that any coefficient C_{ij} is equal to the coefficient C_{ji}. In a material where no particular spacial symmetry exists the values of 21 different quantities must be known in order to define the elastic properties of the medium. However, materials with axes or planes of symmetry allow an establishment of relations between these coefficient and the number of independent coefficients can be further reduced. Thus for a cubic crystal there are only three independent constants. For the case of an isotropic solid the values of the coefficients must be independent of the set of rectangular axes and therefore the generalized form of Hooke's law has just two independent constants left.

These are denoted by λ and μ and replace the initial coefficients as follows:

$$C_{12} = C_{13} = C_{21} = C_{23} = C_{31} = C_{32} = \lambda \tag{38}$$

$$C_{44} = C_{55} = C_{66} = \mu \tag{39}$$

$$C_{11} = C_{22} = C_{33} = \lambda + 2\mu \tag{40}$$

and all the other 24 coefficients become zero.

The generalized form of Hooke's law (equations (32) - (37)) can then be rewritten in

$$\sigma_{xx} = \lambda(\epsilon_{xx} + \epsilon_{yy} + \epsilon_{zz}) + 2\mu\epsilon_{xx} \tag{41}$$

$$\sigma_{yy} = \lambda(\epsilon_{xx} + \epsilon_{yy} + \epsilon_{zz}) + 2\mu\epsilon_{yy} \tag{42}$$

$$\sigma_{zz} = \lambda(\epsilon_{xx} + \epsilon_{yy} + \epsilon_{zz}) + 2\mu\epsilon_{zz} \tag{43}$$

$$\sigma_{yz} = \mu\epsilon_{yz} \tag{44}$$

$$\sigma_{zx} = \mu\epsilon_{zx} \tag{45}$$

$$\sigma_{xy} = \mu\epsilon_{xy} \tag{46}$$

where $(\epsilon_{xx} + \epsilon_{yy} + \epsilon_{zz})$ represents the change in volume of a unit cube and is called the dilatation Δ.

The two elastic constants , λ and μ, are known as Lamé constants and completely define the elastic behavior of an isotropic solid. For the sake of convenience, however, four elastic constants are used. These are Young's modulus E, Poisson's ratio ν, bulk modulus k, and the rigidity modulus which is identical with the second Lamé constant μ. Using equations (41) to (46), E, ν, and k can be expressed in terms of λ and μ:

$$E = \frac{\sigma_{xx}}{\epsilon_{xx}} = \frac{\mu(3\lambda + 2\mu)}{\lambda + \mu} \tag{47}$$

$$\nu = \frac{\lambda}{2(\lambda + \mu)} \tag{48}$$

$$k = \lambda + \frac{2\mu}{3}. \tag{49}$$

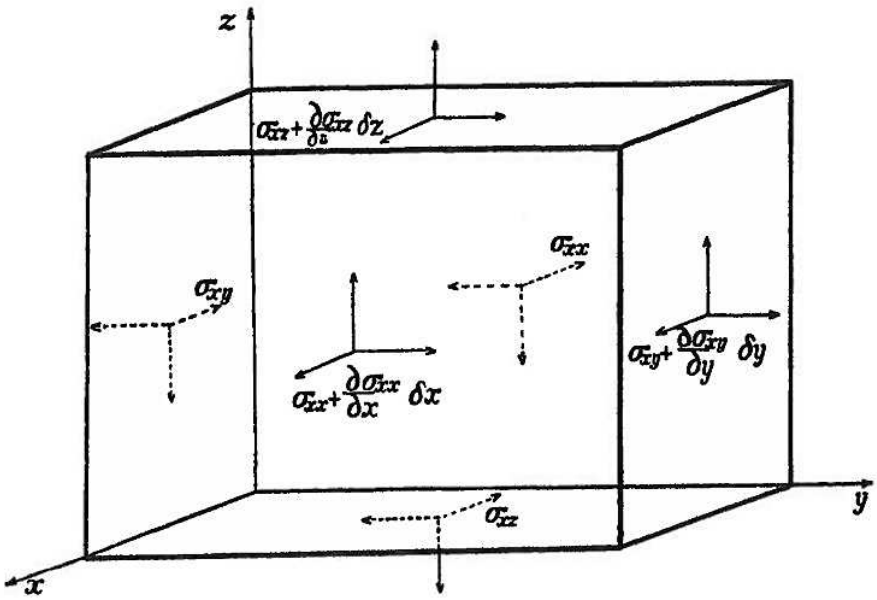


Figure 3: Variation of stress components acting on an infinitesimal rectangular cube

Finally, the shear modulus or rigidity is μ and characterizes the ration between the shear stress and shear strain as given by equations (35) - (37):

$$\mu = \frac{\sigma_{yz}}{\epsilon_{yz}} = \frac{\sigma_{zx}}{\epsilon_{zx}} = \frac{\sigma_{xy}}{\epsilon_{xy}}. \tag{50}$$

Equations of motion in an elastic solid

In order to obtain the equations of motion for an elastic medium the variation in stress across a infinitesimal, equilateral, and rectangular cube with its sides parallel to a set of rectangular axes (see Fig. 3) has to be considered. The components of stress vary across the faces. To approximate the force that is acting on each face, the value of stress at the center of each face is taken and multiplied by the area of the corresponding face. As the figure suggests, six separate forces act along each axis.

For instance, the sum of all forces along the x-axis is

$$\begin{aligned}\sum F_x = &\left(\sigma_{xx} + \frac{\partial \sigma_{xx}}{\partial x}\delta x\right)\delta y\delta z - \sigma_{xx}\delta y\delta z \\ &+ \left(\sigma_{xy} + \frac{\partial \sigma_{xy}}{\partial y}\delta y\right)\delta x\delta z - \sigma_{xy}\delta x\delta z \\ &+ \left(\sigma_{xz} + \frac{\partial \sigma_{xz}}{\partial z}\delta z\right)\delta x\delta y - \sigma_{xz}\delta x\delta y \end{aligned} \tag{51}$$

which simplifies to

$$\sum F_x = \left(\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z}\right)\delta x\delta y\delta z. \tag{52}$$

Including Newton's second law of motion and neglecting body forces such as gravity, the right-hand sides of above equations are equal to

$$\left(\rho\delta x\delta y\delta z\right)\frac{\partial^2 u}{\partial t^2} \tag{53}$$

with the material density ρ and the displacement in x-direction u, so that

$$\rho\frac{\partial^2 u}{\partial t^2} = \frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z} \tag{54}$$

and similarly

$$\rho\frac{\partial^2 v}{\partial t^2} = \frac{\partial \sigma_{yx}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{yz}}{\partial z} \tag{55}$$

$$\rho\frac{\partial^2 w}{\partial t^2} = \frac{\partial \sigma_{zx}}{\partial x} + \frac{\partial \sigma_{zy}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z}. \tag{56}$$

These equations will hold, no matter how the stress-strain behavior of the medium looks like. In order to solve them, the previously defined elastic equations can be applied.

Finally, the general form of the three-dimensional wave equation using index notation is

$$\rho\frac{\partial^2 u_i}{\partial t^2} = \frac{\partial \sigma_{ij}}{\partial x_j} \tag{57}$$

or

$$\rho\frac{\partial^2 u_i}{\partial t^2} = \frac{\partial \left(C_{ijkl}\epsilon_{kl}\right)}{\partial x_j} \tag{58}$$

by plugging in the constitutive relation for linear elastic materials.

2.3 Weak Formulation of Linear Wave Equation

Now that the phenomenon is described and the wave equation is set up, a way needs to be found how the differential equation (57) can be solved.

There are several analytical solutions that can precisely represent the wave field for simple cases, such as homogeneous and isotropic materials. However, for more complex materials that are neither homogeneous nor isotropic an analytical solution is very difficult to find. Therefore, scientists tend to apply numerical methods to solve complicated differential equations. A very common numerical method is the finite element method FEM that is utilized to simulate mechanical, thermal, thermo-mechanical, and electrical problems by solving the specific differential equation. Thereto, the differential equation has to be transformed from the strong form into a weak form. In other words, the conventional differential equation is transformed into an alternate representation of the differential equation and it is called weak since the order is reduced by one. The strong form imposes continuity and differentiability requirements on the potential solutions to the equation, whereas the weak form relaxes these requirements on solutions to a certain extent. This means that a larger set of functions are solutions of the weak form (Gelfand and Fomin [7]).

In order to fully understand how the transformation works and why it is applied, the wave equation (57) will be transformed into its weak form. The first step is to write down the initial strong form of the differential equation. As the equation is to be solved on the entire domain the previously provided form needs to be integrated over the entire volume.

$$\int_V \rho \frac{\partial^2 u_i}{\partial t^2} \, dV = \int_V \frac{\partial \sigma_{ij}}{\partial x_j} \, dV \tag{59}$$

Moving all terms to the left-hand side leads to

$$\int_V \rho \frac{\partial^2 u_i}{\partial t^2} \, dV - \int_V \frac{\partial \sigma_{ij}}{\partial x_j} \, dV = 0. \tag{60}$$

Multiplying both sides of equation (60) by a random, nonzero test function v_i

$$\int_V \rho \frac{\partial^2 u_i}{\partial t^2} \, v_i \, dV - \int_V \frac{\partial \sigma_{ij}}{\partial x_j} \, v_i \, dV = 0. \tag{61}$$

will not change the equality of the equation. Equation (61) can be rewritten as

$$\begin{aligned} \int_V \rho \frac{\partial^2 u_i}{\partial t^2} \, v_i \, dV &= \int_V \frac{\partial \sigma_{ij}}{\partial x_j} \, v_i \, dV & (62) \\ &= \int_V \frac{\partial \left(C_{ijkl} \epsilon_{kl} \right)}{\partial x_j} \, v_i \, dV & (63) \\ &= C_{ijkl} \int_V \frac{\partial^2 u_k}{\partial x_j \partial x_l} \, v_i \, dV & (64) \end{aligned}$$

by assuming

$$\frac{\partial C_{ijkl}}{\partial x_j} = 0 \tag{65}$$

and

$$\epsilon_{kl} = \frac{\partial u_k}{\partial x_l}. \tag{66}$$

Using the integration by parts method for the right-hand side yields

$$\int_V \rho \frac{\partial^2 u_i}{\partial t^2} \, v_i \, dV = C_{ijkl} \left[\int_V \frac{\partial}{\partial x_j} \left(\frac{\partial u_k}{\partial x_l} v_i \right) dV - \int_V \frac{\partial u_k}{\partial x_l} \frac{\partial v_i}{\partial x_j} dV \right]. \tag{67}$$

The first term in the brackets in equation (67) can be converted from a volume integral into a surface integral by exploiting the divergence theorem

$$C_{ijkl} \int_V \frac{\partial}{\partial x_j} \left(\frac{\partial u_k}{\partial x_l} v_i \right) dV = C_{ijkl} \int_S \left(\frac{\partial u_k}{\partial x_l} v_i \right) n_i \, dS, \tag{68}$$

where n_i is the outward normal vector and S the surface of the domain.

Since

$$C_{ijkl} \int_S \left(\frac{\partial u_k}{\partial x_l} v_i \right) n_i \, dS = \int_S C_{ijkl} \left(\frac{\partial u_k}{\partial x_l} v_i \right) n_i \, dS = \int_S \left(\sigma_{ij} v_i \right) n_i \, dS \tag{69}$$

and the applied stress free boundary condition requires

$$\sigma_{ij} n_i = 0 \tag{70}$$

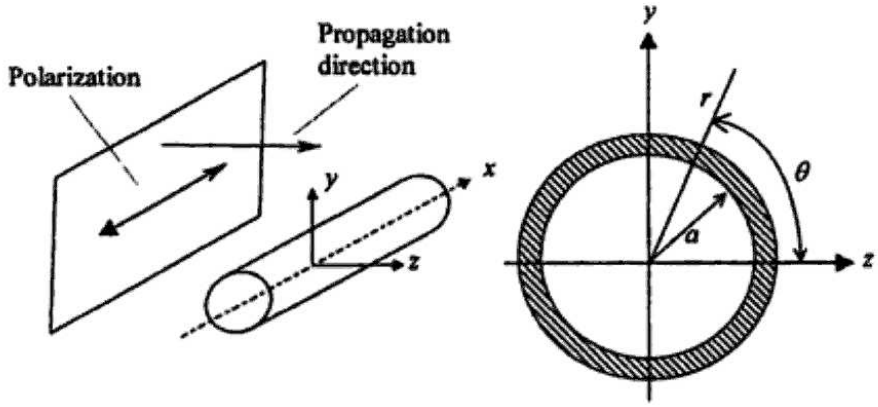


Figure 4: Shear wave scattering from a circular cavity

on the surface S of the domain, the term

$$C_{ijkl} \int_S \left(\frac{\partial u_k}{\partial x_l} v_i \right) n_i \, dS = \int_S (\sigma_{ij} v_i) \, n_i \, dS = 0 \tag{71}$$

vanishes.

Lastly, the weak form of the wave equation is represented by

$$\int_V \rho \frac{\partial^2 u_i}{\partial t^2} \, v_i \, dV = - \int_V C_{ijkl} \, \frac{\partial u_k}{\partial x_l} \frac{\partial v_i}{\partial x_j} \, dV \tag{72}$$

or

$$\int_V \rho \frac{\partial^2 u_i}{\partial t^2} \, v_i \, dV + \int_V C_{ijkl} \, \frac{\partial u_k}{\partial x_l} \frac{\partial v_i}{\partial x_j} \, dV = 0. \tag{73}$$

Clearly, the order of the differential equation could be reduced by one and this form can now be used as input for the numerical solving process.

2.4 Wave Scattering and Attenuation

In order to illustrate the method of scattering theory, scattering of shear horizontal (SH) waves by a cylindrical cavity is considered (see Rose [18]). That is one of the simplest cases because there is only one governing equation of motion for one unknown displacement $u = u_x(y, z)$ in the x-direction, while $u_y = u_z = 0$ (compare Fig. 4).

The equation of motion for this simplified case is

$$\nabla^2 u = \frac{1}{c_T^2} \frac{\partial^2 u}{\partial t^2} \tag{74}$$

or expressed in cylindrical coordinates

$$\nabla^2 u = \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2} + \frac{\partial^2 u}{\partial x^2}, \tag{75}$$

where

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \tag{76}$$

and

$$c_T^2 = \frac{\mu}{\rho}. \tag{77}$$

Considering traction-free boundary conditions on the cavity surface

$$\tau_{rr} = \tau_{r\theta} = \tau_{rx} \quad \text{at} \quad r = a \tag{78}$$

$$\tau_{rr} = \lambda\left(\frac{\partial u_r}{\partial r} + \frac{1}{r}\frac{\partial u_\theta}{\partial \theta} + \frac{u_r}{r} + \frac{\partial u_x}{\partial x}\right) + 2\mu\frac{\partial u_r}{\partial r} \tag{79}$$

$$\tau_{rr} = \mu\left(\frac{1}{r}\frac{\partial u_r}{\partial \theta} + \frac{\partial u_\theta}{\partial r} - \frac{u_\theta}{r}\right) \tag{80}$$

$$\tau_{rr} = \mu\left(\frac{\partial u_x}{\partial r} + \frac{\partial u_r}{\partial x}\right), \tag{81}$$

where λ and μ are the Lamé coefficients. It follows that

$$u_r = u_\theta = 0 \tag{82}$$

and u_x is not dependent on x. Therefore, the obtained boundary condition for the cavity surface is

$$\frac{\partial u}{\partial r} = 0 \quad \text{at} \quad r = a. \tag{83}$$

The incident SH wave (subscript i) propagating in the z-direction is

$$u_i = U_0 e^{i(\omega t - kz)}, \tag{84}$$

where

$$k = \frac{\omega}{c_T}, \tag{85}$$

which satisfies the governing equation of motion (74).

When the incident wave interacts with the cavity, a scattering field will occur. Because of the harmonic character, the scattered wave field (subscript s) can be written as

$$u_s = U_s(r,\theta)e^{i\omega t}. \tag{86}$$

The unknown amplitude $U_s(r,\theta)$ is independent from the x-location and can be seen as a product of $R(s)$ and $S(\theta)$:

$$U_s(r,\theta) = R(r)S(\theta). \tag{87}$$

The resulting total displacement field is given by

$$u(r,\theta)e^{i\omega t} = u_i + u_s \tag{88}$$

and should satisfy equation (74).

Substituting (88) into (74) and using the method of separation of variables as well as the symmetry of results with respect to the x-axis, it follows that

$$S(\theta) = C\cos n\theta \tag{89}$$

and

$$R(r) = A\,H_n^{(2)}(kr) \tag{90}$$

with the Hankel function of the second kind $H_n^{(2)}$ for $k >> 1$

$$H_n^{(2)}(kr) = \sqrt{\frac{2}{\pi kr}}\exp\left\{-i\left(kr - \frac{\pi}{4} - \frac{n\pi}{2}\right)\right\} \tag{91}$$

to satisfy the radiation condition. From equations (87), (89), (90), (86), and by including Bessel functions to solve the problem, the final form of the scattered wave field can be stated as

$$u_s(r,\theta,t) = U_0\sqrt{\frac{2}{\pi kr}}e^{i(\omega t - kr)}\psi_s(\theta), \tag{92}$$

with a known function $\psi_s(\theta)$ and the condition $kr >> 1$.

Scattering in general and the above shown example of scattering produce both magnitude reductions and pulse spreading in dispersive media as a result of wave propagation and interaction with small obstacles or flaws. The pure effect of reduction in wave magnitude, no matter what the cause is, is known as *attenuation*. It can come about from internal friction, energy absorption, energy deviation, or also from geometric effects as in the case of spherical or cylindrical wave propagation.

The most accurate way to take general attenuation into account is to assume an exponential amplitude decay along the propagation direction

$$A(x) = A_0\, e^{-\alpha x}, \tag{93}$$

where $A(x)$ is the amplitude transmitted across some distance x, A_0 the initial amplitude, and α the material specific attenuation coefficient. The attenuation coefficient α describes the fraction of waves that is absorbed, scattered, or simply lost per unit length (one-dimensional) or unit volume (three-dimensional) of the medium. This value essentially accounts for the number of defects per unit length or unit volume of material and the probability of a wave being scattered or absorbed in that specific region.

Finally, the simplified resulting wave field for the one-dimensional case can be written as

$$u(x,t) = A(x)e^{i(kx-\omega t)} \tag{94}$$

$$= A_0\, e^{-\alpha x}\, e^{i(kx-\omega t)}. \tag{95}$$

CHAPTER III

STATISTICAL METHODS

3.1 Microstructure Function

In general, it can be assumed that a realistic microstructure has at least two different length scales that differ from each other by several orders of magnitude. At a lower length scale, a material point in the microstructure is associated with a clearly identifiable local state. As one moves from one location to another in the microstructure, the local states as well as local properties may vary substantially. However, at the higher length scale, a statistically homogeneous sample is expected to show uniform effective properties. These uniform effective properties are often very different from a simple volume average of the local properties at the lower length scale [16].

Consider a heterogeneous material sample Ω from which a group of lower length scale regions $(\Omega^1, \Omega^2, ..., \Omega^j)$ are extracted. The internal structure of each region Ω^j can be described completely by specifying the local state h at each spatial position $x \in \Omega^j$ within the region. The local state h is an element of the local state space H that comprises the complete set of all possible states in the given material system. The local state is typically defined as a combination of several variables. Consequently, if the local state is specified by a set of z parameters, the local state space, H, is a z-dimensional space and each local state h is described by a vector with z components.

Having defined an appropriate local state space of interest for a selected material system, the next task is to quantify the microstructure. The microstructure of each region Ω^j can be accurately captured by the probability density function $m^j(x, j)$, associated with finding the local state h in the probe area around the spatial location

x. Mathematically, this definition of the microstructure function is expressed as [15]

$$m(x,h)dh = \frac{V_h \pm \frac{dh}{2}(x)}{V(x)}, \tag{96}$$

$$\int_H m(x,h)dh = 1, \tag{97}$$

$$\int_H dh = 1, \tag{98}$$

where $V(x)$ denotes the volume of the material probed by the microstructure characterization equipment in the measurement taken at spatial location x, $V_h \pm \frac{dh}{2}(x)$ is the component of $V(x)$ associated with local states that lie within $(h - \frac{dh}{2}, h + \frac{dh}{2})$, and dh is an invariant measure of the local state space. Let the spatial domain of the microstructure be binned into a uniform grid of S cells, whose nodes are enumerated by the ordered tuple $(s1, s2, s3)$ that is represented by the vector s. In an analogous manner, the local state space H be binned into a uniform grid of N discrete local states. For convenience, the same approach is applied that is used for the spatial discretization. Each grid point in the local state space is represented by an ordered discrete state vector n. The resulting discrete microstructure function over the region Q^j is derived from equations (96) - (98) and thus denoted by the form

$$\sum_{n=0}^{N-1} {}^{n}m_s^j = 1, \tag{99}$$

$$0 \leq {}^{n}m_s^j, \tag{100}$$

$$\sum_{n=0}^{N-1} {}^{n}m_s^j = {}^{n}V^j S, \tag{101}$$

where ${}^{n}V^j$ is the volume fraction of the local state n in the region Ω^j and S the product of the node coordinates S_i.

3.2 n-Point Statistics

After the discretization of the domain and its properties, the next logical question is how to characterize the details of the microstructure statistically. Many spatial

descriptive statistics exist and have been utilized successfully. However, for the actual case of describing the spatial distribution of precipitates or essentially describing the distribution of circles in a rectangular domain, the n-point statistics allows the most comprehensive treatment and provides a set of hierarchical measures.

In this section, the 1-point and 2-point statistics are explained corresponding to the discretized microstructure realization ${}^{n}m_s^j$ from the previous section. These first two forms of the n-point statistics are defined as

$$ {}^{n}f^j = \frac{1}{S}\sum_{s=0}^{S-1} {}^{n}m_s^j \tag{102} $$

for the 1-point statistics and

$$ {}^{np}f_t^j = \frac{1}{S}\sum_{s=0}^{S-1} {}^{n}m_s^j {}^{p}m_{s+t}^j \tag{103} $$

for the 2-point statistics.

The expression ${}^{n}f^j$ for the 1-point case is the probability of finding the local state n at a randomly selected point in the region Ω^j and is equivalent to ${}^{n}V^j$ from (101). Basically, it is nothing more than the volume fraction or number density of state n within the specified region.

The latter case of 2-point statistics can be interpreted as the joint probability density of finding local states belonging to n and p at the tail and head of a randomly shaped and oriented vector t thrown into Ω^j. The random vector t is discretized using the same scheme used for the spatial domain of the microstructure. In the k-dimensional case, the vector t is described by the lengths from tail to head along the k coordinate axes (negative lengths possible). Furthermore, it should be noted that there is a tremendous leap in the amount of microstructure information contained in the 2-point statistics compared to the 1-point statistics. The use of 2-point statistics offers many useful statistical measures of the microstructure, such as information on the average shape, size, and spacing of the constituent local states. The most

efficient way to calculate these 2-point statistics is by applying fast Fourier transformation (FFT) methods [14]. Thus, the discrete Fourier transformation (DFT) of the microstructure function ${}^{n}m_s$ (superscript j omitted now for the sake of simplicity) is expressed as

$$ {}^{n}M_k = \Im({}^{n}m_s) = \sum_{s=0}^{S-1} {}^{n}m_s e^{2\pi i\, sk/S} = |{}^{n}M_k|\, e^{i\,{}^{n}\theta_k}, \tag{104} $$

where the term $|{}^{n}M_k|$ is referred to as the amplitude of the Fourier transform, ${}^{n}\theta_k$ as the phase, and k the harmonic index. The discrete Fourier transformation of the 2-point statistical values ${}^{np}f_t$ is computed as

$$ {}^{np}F_k = \Im({}^{np}f_t) = \Im\left(\frac{1}{S}\sum_{s=0}^{S-1} {}^{n}m_s\, {}^{p}m_{s+t}\right) \tag{105} $$

$$ = \frac{1}{S}\, {}^{n}M_k^{*\,p}M_k \tag{106} $$

$$ = \frac{1}{S}\, |{}^{n}M_k|\, |{}^{p}M_k|\, e^{-i\,{}^{n}\theta_k} e^{i\,{}^{p}\theta_k} \tag{107} $$

by exploiting the convolution theorem for two random functions f and g ($*$ is convolution operator)

$$ \Im(f \cdot g) = \Im(f) * \Im(g) \tag{108} $$

and where ${}^{n}M_k^*$ is the complex conjugate of ${}^{n}M_k$. The case $n = p$ in (107) represents a special set of real-valued correlations termed the autocorrelation. The general case of $n \neq p$ leads to cross-correlations and is usually quantified by complex-valued numbers.

3.3 Principal Component Analysis

The central idea of principal component analysis (PCA) is to reduce the dimensionality of a data set which consists of a large number of interrelated variables, while retaining as much as possible of the variation present in the data set. This is achieved by transforming to a new set of variables, the principal components (PC), which are uncorrelated, and which are ordered so that the first few retain most of the variation of the original data.

Assume that $\boldsymbol{x}$ is a vector of p random variables and that the variances of the p random variables and the structure of the covariances between the p variables are of interest. Unless p is small or the structure is very simple, it will often not be very helpful to simply look at the p variances and all of the $\frac{1}{2}p(p-1)$ covariances. An alternative approach is to look for a few derived variables which preserve most of the information given by these variances and covariances.

Although PCA does not ignore covariances, it concentrates on variances. The first step is to look for a linear function $\boldsymbol{\alpha_1^T x}$ of the elements of $\boldsymbol{x}$ which has maximum variance, where $\boldsymbol{\alpha_1}$ is a vector of p constants $\alpha_{11}, \alpha_{12}, ..., \alpha_{1p}$ and ^{T} denotes transpose, so that

$$\boldsymbol{\alpha_1^T x} = \alpha_{11}x_1 + \alpha_{12}x_2 + ... + \alpha_{1p}x_p = \sum_{j=1}^{p} \alpha_{1j}x_j. \tag{109}$$

Next, the linear function $\boldsymbol{\alpha_2^T x}$, uncorrelated with $\boldsymbol{\alpha_1^T x}$, has to be found and so on until a certain number k with $(k \leq p)$ of functions is determined. These variables $\boldsymbol{\alpha_k^T x}$ are the principal components. Up to p principal components could be found but it is hoped that most of the variation in $\boldsymbol{x}$ will be encountered for the first m principal components, where $m << p$.

Having defined principal components, the consequent question is how to find them. Considering the covariance matrix $\boldsymbol{S}$ of the vector $\boldsymbol{x}$ with its (i,j)th elements representing the covariance between the ith and jth element of $\boldsymbol{x}$ when $i \neq j$ and the variance of the jth element when $i = j$, it turns out that for $k = 1, 2, ..., p$ the kth principal component is given by

$$z_k = \boldsymbol{\alpha_k^T x}, \tag{110}$$

where $\boldsymbol{\alpha_k^T}$ is an eigenvector of the covariance matrix $\boldsymbol{S}$ corresponding to its kth largest eigenvalue λ_k. Furthermore, $\boldsymbol{\alpha_k^T}$ is chosen to have unit length, so that the variance of z_k equals the kth eigenvalue

$$\text{var}(z_k) = \lambda_k. \tag{111}$$

CHAPTER IV

NUMERICAL ANALYSIS

4.1 Finite Element Method

The finite element method (FEM) is a numerical method that is very general and powerful in its application to real-world problems that involve complicated physics, geometry, and boundary conditions. The method is endowed with three distinct features that account for its superiority over other competing methods. First, a geometrically complex domain Ω of the problem is represented as a collection of geometrically simple subdomains, called finite elements. Each finite element Ω_j is viewed as an independent domain by itself. In this context, domain refers to the geometric region over which the equations are solved. Second, over each finite element, algebraic equations among the quantities of interest are developed using the governing equations of the problem. Third, the relationships from all elements are globally assembled using certain interelement relationships. Thus, the result for the entire domain can be computed.

Approximations play an important role in many engineering analyses to handle complex problems but are inherently connected with different types of errors. The division of the whole domain into finite elements is one of them. The second stage is when element equations are derived. Typically, the dependent unknowns u of the problem are approximated using the basic idea that any continuous function can be represented by a linear combination of known function ϕ_i and undetermined coefficients c_i

$$u \approx u_h = \sum c_i \phi_i. \tag{112}$$

Algebraic relations among the undetermined coefficients c_i are obtained by satisfying the governing equations, in a weighted-integral sense, over each element. The approximation functions ϕ_i are often assumed as polynomials, and they are derived using concepts from interpolation theory. Therefore, they are termed interpolation functions. Thus, approximation errors in the second stage are introduced both in representing the solution u as well as in evaluating the integrals. Finally, errors are introduced in solving the assembled system of equations. Obvisously, some of the errors discussed above can be zero. When all the described errors are zero, the exact solution of the problem is obtained. Unfortunately , the latter case is not true for most of the actual problems.

4.2 Material Properties

Even though numerical analyses are usually formulated in a very general way and therefore any option, setting or material changes are easily incorporated, the present study focuses on one specific material. The material chosen is a iron-copper FeCu-steel with an amount of about 1 wt% Cu. Copper is of primary importance in the embrittlement of the neutron-irradiated reactor pressure vessel (RPV) steels. This copper has been observed to segregate into copper-rich precipitates within the ferrite matrix under irradiation. Since its role was discovered more than 40 years ago, Cu precipitation in α-Fe has been studied extensively under irradiation as well as under thermal aging using atom probe tomography, small angle neutron scattering, and high resolution transmission electron microscopy.

As the number density of copper precipitates can be an indicator for radiation damage, the exact knowledge about material, measurement methods, and interpretation of measured data is an essential aspect. This study will help to interpret

Table 1: Material properties of FeCu-system components

Material	**Material Property**	**Value**
Fe (matrix)	Pressure wave speed c_p	$5912\,m/s$
	Shear wave speed c_s	$3240\,m/s$
	Mass density ρ	$7874\,kg/m^3$
	Lamé coefficient λ	$1.09894\,10^{11}$
	Lamé coefficient μ	$8.26581\,10^{10}$
Cu (precipitate)	Pressure wave speed c_p	$4760\,m/s$
	Shear wave speed c_s	$2325\,m/s$
	Mass density ρ	$8960\,kg/m^3$
	Lamé coefficient λ	$1.06143\,10^{11}$
	Lamé coefficient μ	$4.84344\,10^{10}$

experimentally measured attenuation coefficients and relate them to important microstructural configuration details. Important mechanical properties of each component of the FeCu system are given in Table 1, which are also used for the present numerical analysis.

4.3 Structure of Analysis

The main goal of the numerical analysis is to show the existence of and to find a quantification of correlations between microstructural defects, in this case precipitates, and the material attenuation, by performing a statistical analysis. A statistical analysis implies the assessment of a large number of random microstructures. These random microstructures are generated virtually and evaluated with numerical and statistical methods to determine if there is a causal link between microstructural configuration and acoustic properties.

The present section is divided into three parts. In the first, the generation of the microstructure is described, and which exact properties are varied throughout the analysis. The second part explains every detail of how the microstructure is evaluated, how the wave equation is solved, and how the acoustic properties are computed. Finally, the third and last part links the results and shows how microstructural and acoustic properties are correlated.

Table 2: Initial material information for microstructure generation

Precipitate Phase Property	**Value**
Number density ϕ	0.88 %
Total number N	200
Shape	Circular
Mass density ρ	Constant
Spatial arrangement	Random/ Statistically constant
Size/ Radius r	Random/ Statistically constant
Stiffness	Random/ Statistically constant

4.3.1 Microstructure Generation

The microstructure is assumed to have an initial monocrystalline microstructure with absolute constant material properties, that are number density of precipitates, total number of precipitates, shape of precipitates, stiffness of lattice structure, and density of precipitates and lattice structure (see Table 2). Besides that, the arrangement, individual size, and stiffness of the precipitate phase is random but still statistically constant. By statistically constant, it is assumed that these three precipitate characteristics (arrangement, size, and stiffness) show a normal distribution with a standard deviation σ around the mean value μ. So, statistically, on a large length scale, the stated characteristics will be globally constant, but on a much smaller length scale, a local variation of those values is clearly detectable. To satisfy the requirements of a statistical analysis, many samples have to be taken into account ($>>$ 100). The higher the number of linked microstructures, the more accurate the information on average acoustic properties and their respective microstructural properties. In this study, the number of microstructures generated is usually 200, depending on mesh density and time step size. Furthermore, as a first step in this complex network, only the precipitate arrangement or their spatial distribution, respectively, is varied while keeping the size and stiffness of the precipitates constant.

Now, the microstructures are virtually generated, with only the spatial distribution as random in this first case. In other words, N precipitates of size or radius r are

randomly distributed in a squared two-dimensional domain of edge lengths l while satisfying the number density constraint. The number density ϕ and the edge length l of the domain considered are redundant and related by

$$\phi = \frac{N\pi r^2}{l^2}. \tag{113}$$

Therefore, the domain size or the length l, respectively, is automatically determined by defining ϕ, N, and r.

Since in reality precipitates cannot intersect, a compatibility check has to be performed after generating the random positions of the precipitates. The minimum distance between two precipitate centers equals the sum of their radii:

$$d_{ij} \geq r_i + r_j \tag{114}$$

or for the current case of no variation in the precipitate size ($r_i = r_j$)

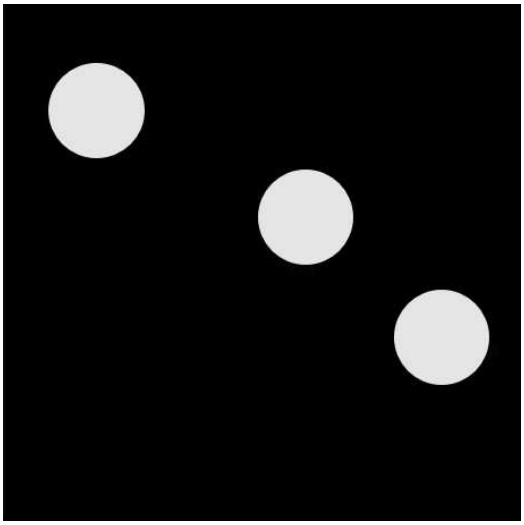
$$d_{ij} \geq 2r_i. \tag{115}$$

Another important aspect of this domain is its periodicity. Periodicity in this case means that a string of several identical microstructures forms a new, continuous microstructure. That means the left and right as well as the upper and lower boundary of a single domain show identical microstructural details. This is what makes a row of them continuous. However, due to practicality and compatibility, no precipitates are located on the boundary. In other words, precipitates are not cut off by the domain boundary.

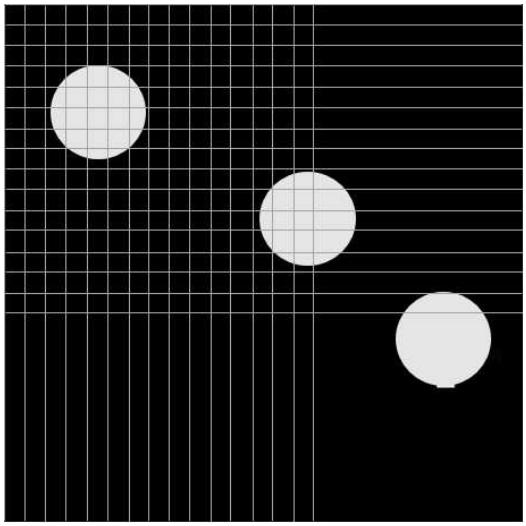
Once these constraints are fulfilled, the microstructure is completely defined and needs to be discretized to allow further numerical evaluations. Discretization in this context concerns the process of transferring the continuous model into discrete counterparts. The discretization method that is applied for the present microstructure is called voxelization. Voxelization (rasterization), as stated before, is concerned with converting geometric objects from their continuous geometric representation into a

set of voxels that best approximates the continuous object [5]. In 2D voxelization the squared voxels (pixels) are directly drawn onto the surface (see Figure 5(b)) to be visualized, and eventually filtering is applied to reduce the aliasing artifacts. The example in Figure 5 does not represent the previously introduced microstructures (Table 2) but is chosen to explain the voxelization in a more comprehensive way. For the current case of a two-phase structure, the voxels are either assigned as precipitate or as lattice structure (not precipitate). This twofold situation is very well suited to be represented by a binary system (also called digitization) or for the two-dimensional case by a binary matrix where 1 indicates precipitates, 0 no precipitates, and the index of each entry the location on the voxelized surface (see Figure 5(d)). For visualization purposes, as in Figure 5(c), the color white is used to represent precipitates (binary value 1) and black to show the lattice structure (binary value 0). The higher the voxel density the higher the accuracy of the voxelized representation of the original structure. The previous statement is true for low voxel densities, however voxel densities that exceed a certain threshold do not lead to an increased accuracy but to a disproportionately high computational cost. Therefore, voxelized structures that allow to identify major microstructural characteristics are sufficient for the current study since only two different states are observable.

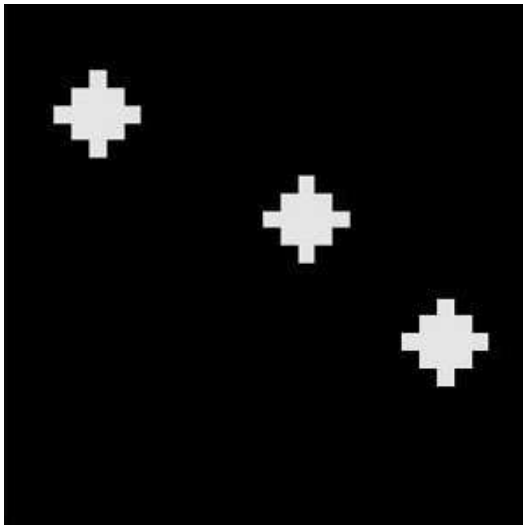
The previous step yielded the discretized, as well as the digitized form of the initial, two-dimensional microstructure with precipitates. To characterize the precipitate distribution and individual size of precipitates within each microstructure, the statistical method of n-point statistics or two-point statistics, respectively, is applied. The latter case of two-point statistics can be interpreted as the joint probability density of finding local states at the tail and head of a randomly shaped and oriented vector t thrown into the microstructure domain. Figure 6 visualizes the two-point statistical results of the microstructure shown in Figure 5(a). In Figure 6, the axes or the coordinates, respectively, indicate the shape and orientation of the random vector



(a) Original microstructure



(b) Microstructure voxelization

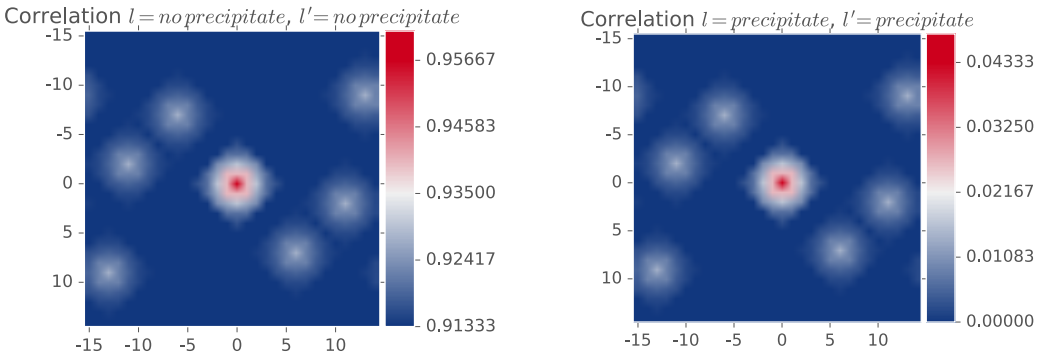


(c) Voxelized microstructure

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	1	1	1	0	0	0	0
0	0	1	1	1	1	1	0	0	0
0	0	0	1	1	1	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0

(d) Digitized microstructure

Figure 5: Microstructure voxelization and digitization



(a) (b)

Figure 6: Two-point statistics of microstructure in Figure 5(a)

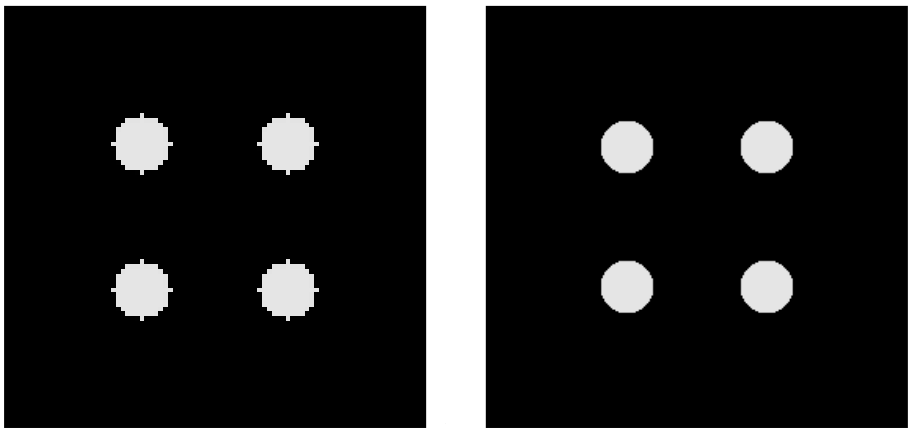
t for the two-dimensional case. The abscissa describes the horizontal and the ordinate the vertical length of t, whereas the colors red and blue represent the probability of finding distinct states at both ends of the vector t. Figure 6(a) shows the autocorrelation of the state *no precipitate*. In particular, it is the probability to find the previously mentioned state *no precipitate* at both ends of the random vector t with its shape and orientation defined by the x- and y-position of the specific probability. The expected value of probability is illustrated by the color of the point. For example, the vector t with coordinates $(x, y) = (0, 30)$ is expected to have a probability of about 92% to have the state *no precipitate* at its head and tail. Another interesting instance is the zero vector $t = (0, 0)$ that traces back to the one-point statistics and is nothing else than the average probability to find a distinct state within the regarded domain.

Comparing Figures 6(a) and 6(b), it occurs that both have the same probability distribution, however with different values as the colorbars indicate. The reason is that the autocorrelation of the state *precipitate* (both ends of the vector t are located in the state *precipitate*) is the exact complementary event of the autocorrelation for the state *no precipitate* since the current case only includes the two different states *precipitate* and *no precipitate*. Therefore, the information contained in Figures 6(a) and 6(b) is redundant for the case of exactly two distinct states and can be expressed

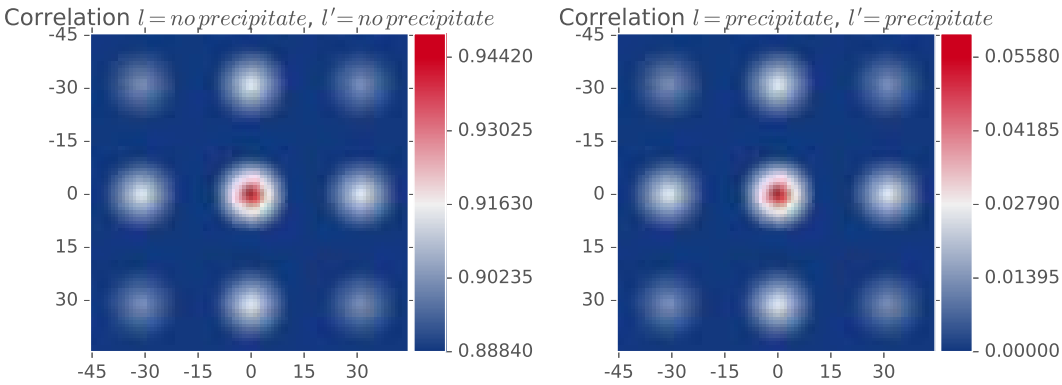
by a single autocorrelation plot.

Furthermore, the two-point statistics not only provide knowledge about the pure location or spatial distribution of the precipitates, but also knowledge about their physical shape. Focusing on the center of the two-point statistics plot in Figure 6(a), it clearly unveils similarities with the initial microstructure's individual precipitate shape in Figure 5(a). High valued expectations are found in the middle of the star-shaped entity with decreasing probabilities towards the tips. The tips have lower values since random vectors, t with a comparatively high length $|t|$ are less likely to fit completely in the voxelized precipitate. Also, as precipitate shape and size do not change throughout the study, the entities in the two-point statistics plot exactly match the actual physical appearance of the precipitates. Otherwise those entities would be a representation of the average size and shape of all precipitates occurring.

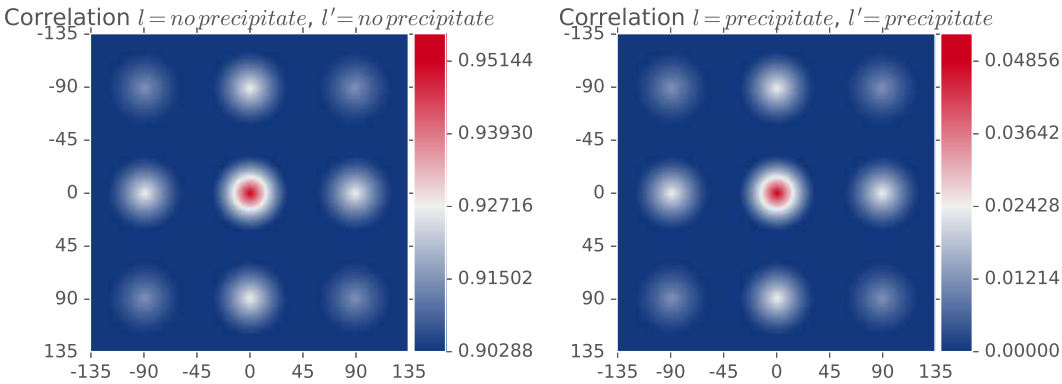
Moreover, the quality of information that such a two-point statistical evaluation includes is highly determined by the goodness of voxelization. The higher the voxel density the higher the accuracy of the discretized representation and two-point statistics of the microstructure, shown in Figure 7. By increasing the voxel density the actual voxel size decreases and entities within the domain can be represented more precisely. First, the voxelized representation of a circle looked more like a star (Figure 5(c)) with 900 voxels for the entire domain. Increasing the number of voxels to 8100 (Figure 7(a)) and 72900 voxels (Figure 7(b)) leads to much more realistic and representative forms than before. Not only does the voxelized version of the microstructure shows a more circular shape of the precipitates, but also the two-point statistics plot is obviously less noisy and the defined number density of precipitates can be represented more accurately. However, increasing the voxel density over a certain threshold value is not necessarily desirable, as the influence of increasing computational cost gets much bigger than the actual accuracy gain. The number of voxels chosen for this study is $600^2 = 360000$ as it is assumed to be very close to the threshold mentioned.



(a) Voxelization with 8,100 voxels (b) Voxelization with 72,900 voxels



(c) Two-point statistics of microstructure 7(a)



(d) Two-point statistics of microstructure 7(b)

Figure 7: Voxelization and two-point statistics accuracy in dependency of the voxel density

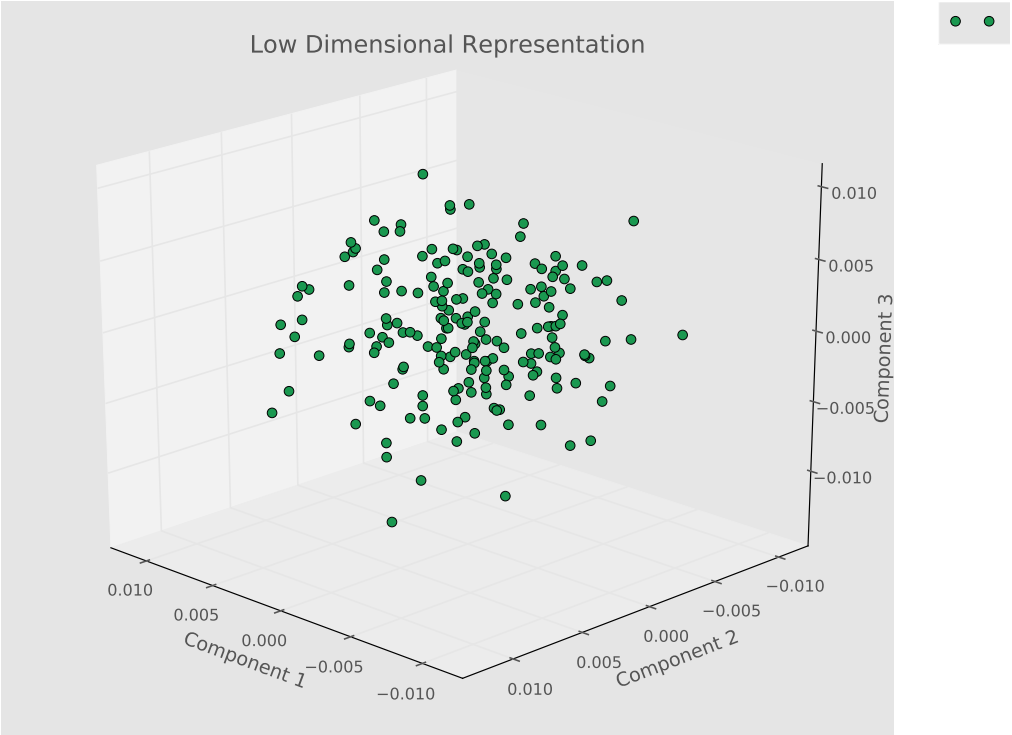


Figure 8: PCA of two-point statistics, 200 random microstructures

Once the two-point statistics is calculated for each microstructure generated, the data size and dimensionality have risen by a large amount. As the main goal is to relate microstructural to acoustic properties, a way has to be found to easily characterize and distinguish between each microstructure. A possible and effective way to do so is the statistical method principal component analysis (PCA). The central idea of principal component analysis is to reduce the dimensionality of a data set which consists of a large number of interrelated variables, while retaining as much as possible of the variation present in the data set. This is achieved by transforming to a new set of variables, the principal components, which are uncorrelated, and which are ordered so that the first few retain most of the variation of the original data.

Figure 8 shows a PCA plot that reduces the dimensionality to three or to three

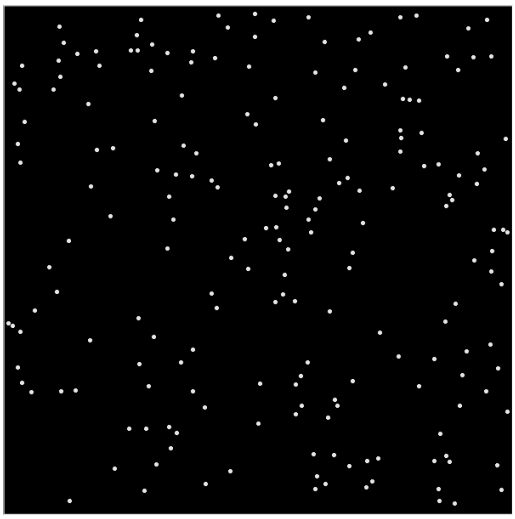


Figure 9: Voxelized microstructure with random distribution of 200 precipitates

principal components. Each dot in the PCA plot represents a distinct microstructure by relating its spatial position to the first three principal values. Thus, the microstructures and their physical precipitate appearance are statistically characterized and can be clearly distinguished. A sample of a voxelized microstructure with a random distribution of 200 precipitates and a discretization with 360,000 voxels, as it is used for the current analyses, is displayed in Figure 9. This is the basis for the next task, the microstructure evaluation or determination of the acoustic properties, respectively, of each test object.

4.3.2 Microstructure Evaluation

To evaluate the acoustic properties, that is the attenuation behavior and material nonlinearity, the wave equation (116) has to be solved for each microstructure.

$$\rho \frac{\partial^2 u_i}{\partial t^2} = \frac{\partial \sigma_{ij}}{\partial x_j} \tag{116}$$

Since closed form analytical solutions for such complex geometries are generally unavailable, numerical methods have to be applied to approximate the exact solution of the wave equation. Since computers work with finite memories and perform only

finite calculations, approximations must be made in order to solve the wave equation numerically. The numerical method used in this study is the finite element method (FEM) that discretizes the continuous domain. These approximations have to deal with issues concerning the discretization of space and time, methods of solving the discretized version of the equation, and error analysis.

Spatial and time discretizations are related through the Courant-Friedrichs-Lewy (CFL) condition (117), a necessary condition for convergence while solving the hyperbolic partial differential wave equation

$$CFL = \frac{c\,\Delta t}{\Delta x}, \tag{117}$$

where c is the magnitude of the wave speed, Δt the time step, and Δx the element length of the discretized domain. Depending on whether the method to solve the discretized equation is explicit or implicit, the optimal CFL-number changes. In this study, an explicit central difference [12]

$$U'(x_i) = \frac{U(x_{i+1}) - U(x_{i-1})}{2h} \tag{118}$$

is utilized for the spatial derivative, where h corresponds to Δx and denotes the distance between two node points. The time integration on the other hand is approached with the implicit multi-step Bathe-method [1]. In Bathe's method, the complete time step Δt is subdivided into two equal sub-steps. For the first sub-step the trapezoidal rule is used and for the second sub-step the 3-point Euler backward method is employed with the resulting equations

$$^{t+\frac{\Delta t}{2}}\dot{U} = {}^{t}\dot{U} + \frac{\Delta t}{4}\left({}^{t}\ddot{U} + {}^{t+\frac{\Delta t}{2}}\ddot{U}\right) \tag{119}$$

$$^{t+\frac{\Delta t}{2}}U = {}^{t}U + \frac{\Delta t}{4}\left({}^{t}\dot{U} + {}^{t+\frac{\Delta t}{2}}\dot{U}\right) \tag{120}$$

$$^{t+\Delta t}\dot{U} = \frac{1}{\Delta t}{}^{t}U - \frac{4}{\Delta t}{}^{t+\frac{\Delta t}{2}}U + \frac{3}{\Delta t}{}^{t+\Delta t}U \tag{121}$$

$$^{t+\Delta t}\ddot{U} = \frac{1}{\Delta t}{}^{t}\dot{U} - \frac{4}{\Delta t}{}^{t+\frac{\Delta t}{2}}\dot{U} + \frac{3}{\Delta t}{}^{t+\Delta t}\dot{U}, \tag{122}$$

Table 3: Discretization properties

Discretization Property	**Method / Value**
Discretized derivatives in space	Central difference (explicit)
Discretized derivatives in time	Bathe method (implicit)
Element type	Linear, triangular (2D)
Mesh density	30 elements/wavelength
CFL-number	1.0

where U, $\dot{U}$, and $\ddot{U}$ represent the approximation of the exact displacements and their derivatives u, $\dot{u}$, and $\ddot{u}$.

Since the input wave equation for the finite element solver is in its weak form

$$\int_V \rho \frac{\partial^2 u_i}{\partial t^2} \, v_i \, dV + \int_V C_{ijkl} \, \frac{\partial u_k}{\partial x_l} \frac{\partial v_i}{\partial x_j} \, dV = 0, \tag{123}$$

only first order spatial derivatives are necessary for the displacements u_i and the the test functions v_i. Substituting all continuous expressions by discrete formulations, the problem can be solved with a finite element solver. In this specific study, the partial differential equation solver *FreeFEM++* is applied with a conjugate gradient (CG) method and a CFL-value of 1.0 (see Table 3).

As the case here is just two-dimensional and the order of spatial derivatives is reduced to one through the weak formulation, linear triangular finite elements are sufficient to represent the displacements in an accurate manner. However, the appropriate size of those elements is not determined yet. In order to find a well-suited element size, a convergence analysis has to be performed by incrementally reducing the element size or increasing the mesh density, respectively. Usually, an increased mesh density leads to a higher accuracy of results until a certain threshold is reached. After passing that threshold, an increase in mesh density will not cause but marginal difference of the results (displacements in this case). The actual convergence analysis is using 10, 15, 20, 25, 30, and 35 elements per wavelength λ. The coarsest mesh has 10 elements per wavelength and the finest 35. To show that the convergence is independent of the chosen frequency, two normalized frequency cases $ka = 0.25$ and

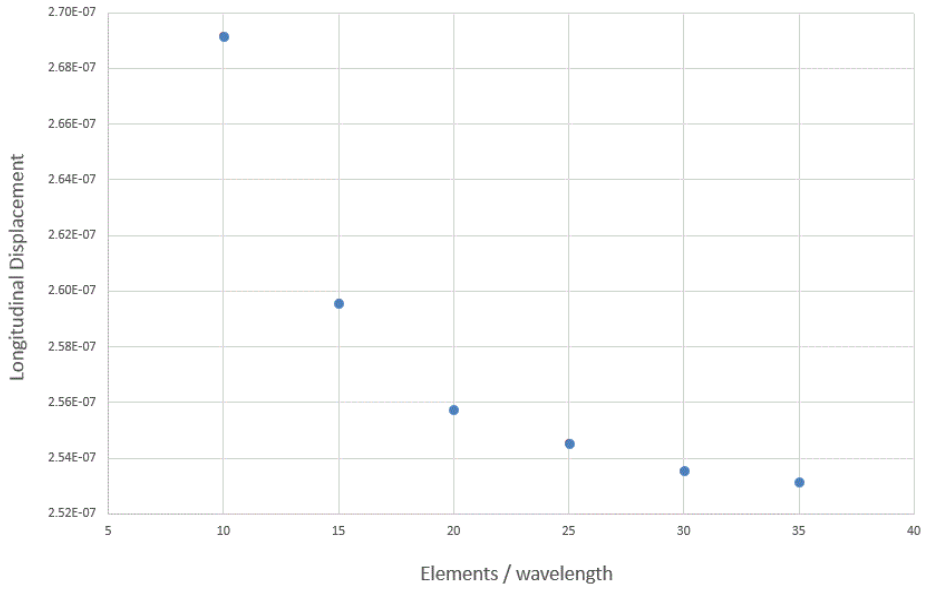


Figure 10: Displacement convergence analysis for $ka = 0.25$

$ka = 0.5$ are investigated, which are shown in Figures 10 and 11, respectively. Both cases unveil converging displacements for increasing mesh densities and it can be assumed that mesh densities higher than 35 elements per wavelength do not improve accuracy. It shows that the convergence limit (position of setting in convergence) is not dependent on the absolute excitation frequency ka and therefore the mesh density can be chosen similarly for all different (normalized) frequencies. Furthermore, an assumed convergence limit of 35 elements per wavelength allows calculations with mesh densities of 30 elements per wavelength with less than 1% uncertainty. Paired with acceptable calculation times, 30 elements per wavelength are set as the standard mesh density or element size, respectively, for all discussed frequency cases.

Once all parameters are determined, the domain needs to be meshed. The previously set standard mesh density is defined on the edges of the squared domain and is increasing towards to middle, towards the precipitates. The precipitates are meshed separately with a slightly smaller element size in order to take their relatively small size correctly into account. Within the discretized precipitate boundaries different material properties such as stiffness can be defined, since in real case the precipitate

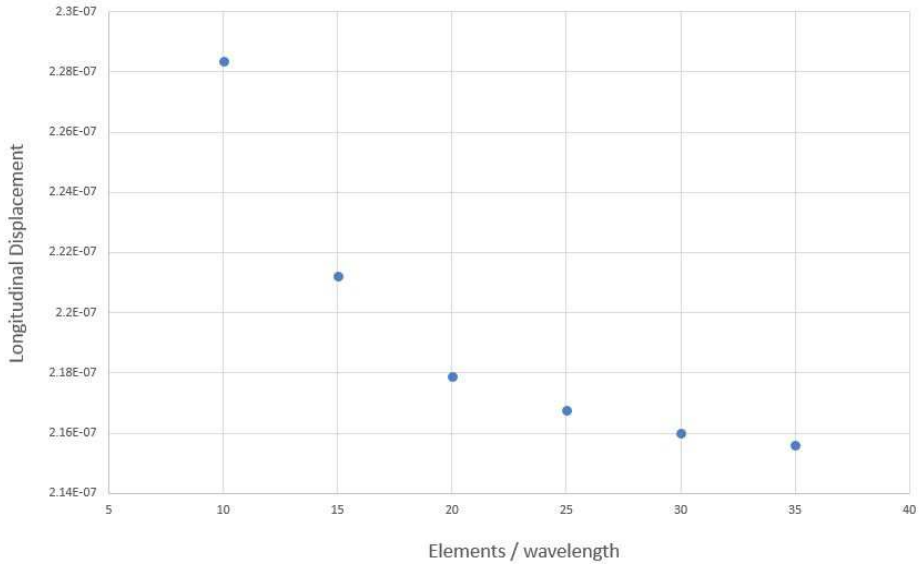


Figure 11: Displacement convergence analysis for $ka = 0.5$

material (Cu) differs from the adjacent lattice material (Fe). Intersecting precipitates are physically impossible and virtually avoided by checking minimum distances between generated precipitates in the previous microstructure generation step. An enlarged sample mesh of a microstructure with 200 randomly arranged precipitates is presented in Figure 12. The green line on the right end is the domain boundary where the boundary conditions are applied on. Precipitates are marked as colored circles and the black lines represent the edges of the triangular elements. In addition, it clearly shows the changing mesh density from the boundary on the right to the colored precipitates.

The last step before the finite element solver can be launched is to define the boundary conditions. As the name says, boundary conditions describe the results' behavior on the domain edges. Assume a two-dimensional coordinate system with its x-axis pointing to the right and its y-axis pointing upwards and the squared microstructure domain's edges parallel to the axes. The left and right boundaries or edges (along the y-axis), respectively, are set to have a periodic boundary condition. That means, displacements and at least their first spatial derivative are symmetric

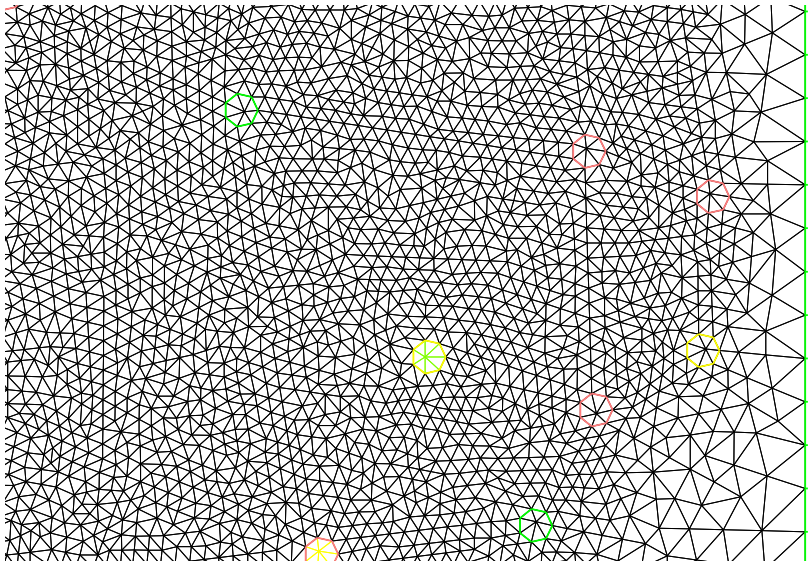


Figure 12: Meshed microstructure (enlarged)

at both boundaries. Moreover, the bottom edge (x-axis or $y = 0$) is said to be the excitation edge and is continuously excited in y-direction with a harmonic function

$$A(t, x, y = 0) = \hat{A}_0 \sin(\omega t) \quad \forall\, x. \tag{124}$$

Thus, a plane wave is propagating in positive y-direction. The remaining, upper edge is basically a free end but is actually treated as a non reflecting boundary by stopping the simulation as soon as the wave reaches that upper edge. The simulation time is therefore determined by the domain edge length l and the fastest wave speed, the p-wave speed c_p:

$$t_{sim} = \frac{l}{c_p}. \tag{125}$$

Now, all parameters are specified and the simulation can start to solve for the displacement field or wave field, respectively. Because the domain is discretized in distinct nodes that are connected to triangular elements, the resulting displacement field is also discrete and consisting of nodal displacement values. Displacements for positions that do not exactly lie on a node are linearly interpolated between nodes of the particular element. Thus, the displacement field is continuous but not continuously differentiable. A visualization of the resulting wave field is given in Figures 13 and

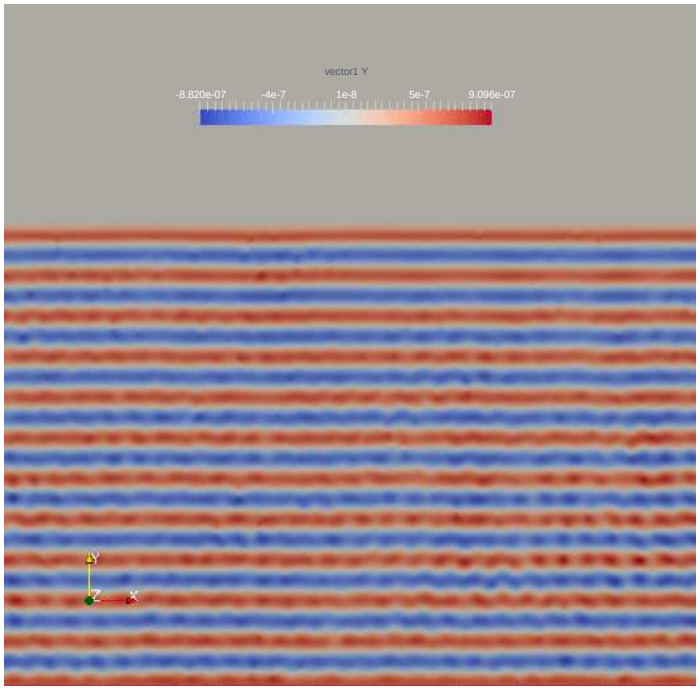


Figure 13: Two-dimensional p-wave field

14, where the wave front has not reached the upper edge yet. The colors indicate the displacements in y-direction for the p-wave and in x-direction for the s-wave. As a plane wave is excited at the excitation edge, the p-wave field always exists, no matter what. The shear wave field, however, only exists if defects are present, such as the introduced precipitates in the actual study. These precipitates are responsible for the scattering effect and thus responsible for shear wave generation and the distortion of the initial wave field. Initial lines of same displacement in Figure 13 get successively noisy as they travel through the material. Not only the dispersivity but also reflections make lines that are close the excitation edge look very odd since that is the location with maximum interference of all reflected waves. Besides, shear waves are definitely existent but not of major importance in this work. They are generated as the p-waves scatter with angles other than 90°.

With that, all needed information is now provided to reach the main goals of this section, namely the determination of the attenuation coefficient α and the nonlinearity

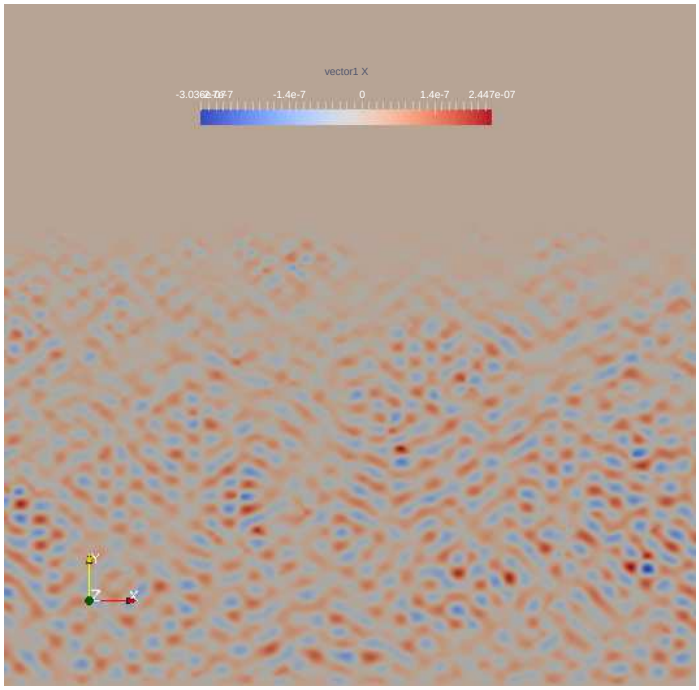


Figure 14: Two-dimensional s-wave field

parameter β.

Attenuation coefficient α

Harmonic wave propagation in viscoelastic materials involves an attenuation of the wave, or in other words a reduction of the displacement amplitude, as it propagates due to the conversion of mechanical energy into heat. However, in this study only the wave propagation in heterogeneous elastic materials is considered. Since elastic materials are completely conservative, the question is how there can be an attenuation effect when mechanical energy is conserved. The answer is that the total mechanical energy must be conserved within the elastic media domain, however, there can be an apparent attenuation effect from incoherent scattering due to the random heterogeneity or the randomly distributed precipitates in this study, respectively. The random nature of the precipitate distribution variation is the essential ingredient to encounter the attenuation effect. In contrast, there are apparent scattering effects in heterogeneous and in all length scales periodic media, but the scattered wave remains

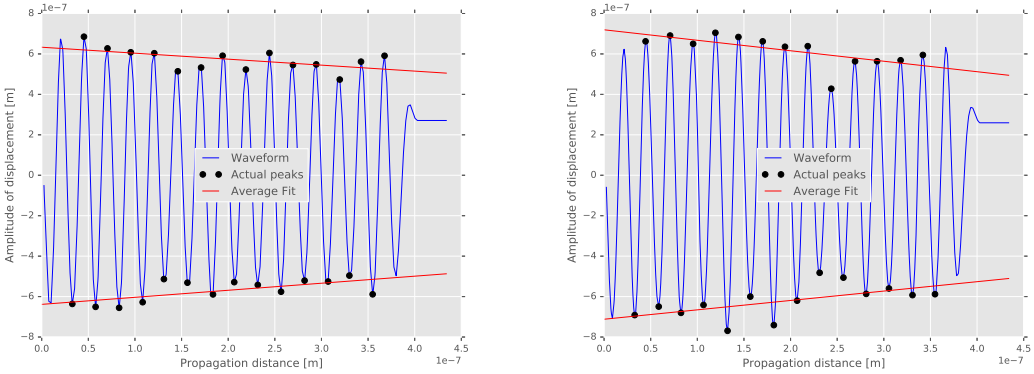
coherent, because of the periodicity of the medium, and thus no attenuation effect is observed.

The determination or measurement of the attenuation behavior can reveal many microstructural properties or even property changes. Nondestructive-evaluation techniques exploit this attenuation behavior information to reveal microscale material damages and predict material life expectancies. To quantify the attenuation behavior, a general assumption of an exponential amplitude decay along the propagation axis is made

$$A(y) = \hat{A}_0 \, e^{-\alpha y}, \tag{126}$$

where $\hat{A}_0$ is the initial excitation amplitude, y the propagation distance along the y-axis, and α the attenuation coefficient. The attenuation coefficient α is dependent on both the material type (iron, copper, aluminum, ...) and the material configuration, where the material configuration is of main interest in this study.

In order to compute α for each virtually generated microstructure, the calculated displacements are fitted to the general attenuation function (126). Thereto, the peaks in the wave field of the very last time step along the y-axis for specific x-positions are considered, as shown in Figure 15. Most of the shown data is obtained through interpolation, only few displacement peaks are actually spatially represented by a node. Having determined the value and spatial position of the peaks, they can be now fitted to (126). The resulting exponential attenuation fit, indicated by red lines in Figure 15, might raise two questions: why are the lines for an exponential fit straight and why are the amplitudes not decreasing monotonically. The answer to the first question is the domain size. Since the regarded propagation distance is relatively small (less than $500nm$), the exponential function line seems to be linear although it is not. For the second question, remember the current material: a heterogeneous lattice-precipitate composite. Some waves might get reflected at out-of-line precipitates and cause interference at a point where lower amplitudes would have been expected. This



(a) $x = 0.25\,l$, $\alpha = 1.29m^{-1}$ (b) $x = 0.75\,l$, $\alpha = 1.36m^{-1}$

Figure 15: Displacement peaks in wave field along y-axis and given x-position

is the case as it occurs in reality, the monotonic decrease is just an ideal assumption.

Due to the random spatial distribution of precipitates, the amplitude decay along the propagation axis y is not the same for every x-position, compare Figures 15(a) and 15(b). In other words, imagine straight lines at different x positions and parallel to the y-axis. The displacements along those lines are not identical and therefore the amplitude decay or the attenuation coefficient α, respectively, is dependent on the spatial position of the chosen line. In order to have one distinct attenuation coefficient for each microstructure, all computed attenuation coefficients per microstructure are combined to an average attenuation coefficient. The present study encompasses 100 equally spaced lines with numerical measurements of the attenuation coefficient to compute the final average attenuation coefficient that will be representative for the respective microstructure.

Not only every line in one microstructure yields different attenuation coefficients but also every entire microstructure will generally reveal distinct average attenuation coefficients. Due to this fact, that is mainly caused by the random precipitate distribution, the results of the statistical analysis will show a standard distribution with a standard deviation σ around the overall mean value μ. An example of such a standard distribution for an analysis of 200 random microstructures and the above presented

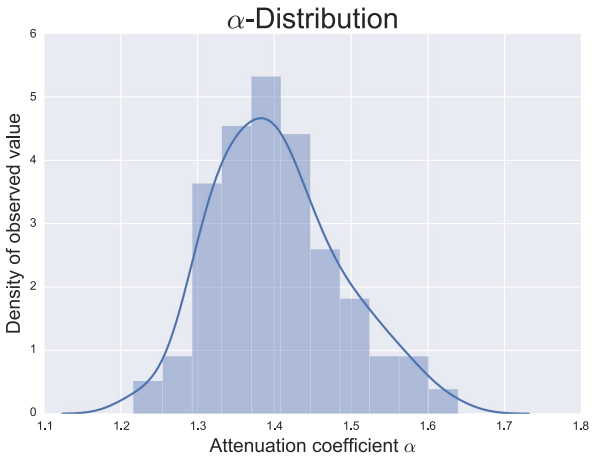


Figure 16: Distribution of average attenuation coefficients

normalized frequency case of $ka = 0.5$ is shown in Figure 16. Interpretations and conclusions of this appearance can be found in the next chapter.

4.3.3 Coupling Microstructural and Acoustic Information

The methods that have been introduced in the last two sections are capable of generating and quantitatively describing the microstructures as well as determining their acoustic properties, while all these methods are utilized virtually. As the main goal of this study is to find correlations between microstructural and acoustic properties, both sides have to be coupled. More precisely, on the one hand the first three components of the principal component analysis of the two-point statistics and on the other hand the average attenuation coefficient of each microstructure are linked together. By linking, literally the concatenation of the principal component matrix (200 by 3) and the attenuation coefficient vector (200 by 1) is understood. The result is a 200 by 4 matrix with the first three columns representing the principal components and the last column the average attenuation coefficient. Now, the two sides of information are linked but the question if there is a correlation between them is still not answered. To do so, a second and last principal component analysis on the generated linked data matrix is applied. Again, the original data is projected along axes with the highest

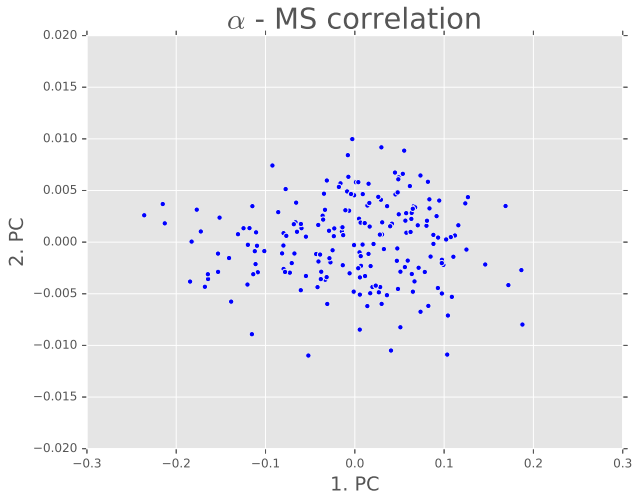


Figure 17: PCA of linked data matrix

variance and the data dimension for this application is reduced to two. Therefore, if range of the first principal components is greater than the range of the second principal components, a certain type of correlation might exist. Figure 17 illustrates the results of this second principal component analysis, where each dot represents a coupled information. The closer those points are located to the first principal axis, the higher the correlation between the analyzed entities. The reason for that is the underlying variance in the data. The smaller the variation along the principal axes other than the first, the data basically changes only in one direction, thus the variables are clearly related. Figure 17 suggests a correlation of the data, however the quality of correlation is not particularly high.

In summary, the methods introduced and described in the last three sections are fully capable of performing a powerful statistical analysis. In addition to that, all these methods are implemented in a Python script that automatically sweeps through all these steps and is very flexible in accepting and changing analysis options and properties. An extract of that Python script and the used FreeFEM++ finite element algorithm can be found in the appendix. The results of the performed statistical analyses and their appraisal follow in the next chapter.

CHAPTER V

NUMERICAL RESULTS

The previously introduced methods microstructure generation, microstructure evaluation, and coupling of microstructural and acoustic information are now practically applied using Python to virtually apply all the introduced methods and FreeFEM++ to solve the wave equation numerically. As the precipitate size is held constant, a variation of the excitation frequency will inevitably change the resulting wave field. Since the wave speed c

$$c = \lambda f \tag{127}$$

is constant in each case, an increasing frequency f has a decreasing wavelength λ as a result and vice versa. For the sake of comprehensibility, the normalized frequency ka

$$ka = \frac{\omega\, a}{c} = \frac{2\pi f\, a}{\lambda f} = \frac{2\pi\, a}{\lambda} \tag{128}$$

is introduced, where k is the wave number and a the precipitate radius.

Four cases are covered in this work: the normalized frequencies $ka = 0.25, 0.5, 0.75,$ and 1.0. The absolute frequencies and additional information related to each case are provided in Table 4.

The four distinct cases are chosen since the most scattering effects are expected

Table 4: Calculation cases for numerical analysis

Case	**1**	**2**	**3**	**4**
Norm. frequency ka	0.25	0.5	0.75	1.0
Abs. frequency f	$1.176 \cdot 10^{11}$ Hz	$2.352 \cdot 10^{11}$ Hz	$3.529 \cdot 10^{11}$ Hz	$4.705 \cdot 10^{11}$ Hz
Abs. wavelength λ	$5.026 \cdot 10^{-8}$ m	$2.513 \cdot 10^{-8}$ m	$1.675 \cdot 10^{-8}$ m	$1.257 \cdot 10^{-8}$ m
Precip. radius a	$2 \cdot 10^{-9}$	$2 \cdot 10^{-9}$	$2 \cdot 10^{-9}$	$2 \cdot 10^{-9}$
λ/a	25.13	12.57	8.38	6.28

to be observed for $0 < ka \leq 1.0$, according to Kim [10]. Higher normalized frequencies tend to have too small wavelengths to have a major influence on the scattering and attenuation behavior of the material. Moreover, high valued normalized or absolute frequencies, respectively, are associated with very dense and fine finite element meshes with a very high number of degrees of freedom. Therefore, calculations with frequency cases $ka > 1.0$ either run out of memory or take computation times that are not compatible with large statistical analyses. The following sections will first illustrate the variation of the attenuation coefficients for each case to show the existence of a correlation between microstructural and acoustic properties. Second, the latter correlation is quantified by performing the second principal component analysis followed by a discussion of the presented results.

5.1 Variation of Data

This first section is concerned with the first part of the statistical analysis results, the variation of the attenuation coefficient. As stated before, each microstructure is related to a distinct attenuation coefficient α. If there is a correlation between the spatial distribution of the precipitates and the attenuation coefficient, a variation of the precipitate distribution has to have an influence on the attenuation behavior. Even if the exact impact is unknown, there has to be a noticeable change of results. To find out whether or not α is sensitive to the spatial distribution of the precipitates, all α-values are depicted in histograms. Figures 18, 19, 20, and 21 depict the histograms for all four cases in the form of histogram combined with a kernel density estimation (KDE). A kernel density estimation is a non-parametric way to estimate the probability density function of a random variable. The optimal number of bins and their size h for the histogram is determined by the Freedman-Diaconis rule

$$h = 2\frac{IQR}{n^{1/3}}, \tag{129}$$

where IQR is the interquartile range and n the number of data points. The interquartile range is a measure of variability, based on dividing a data set into quartiles. Quartiles divide a rank-ordered data set into four equal parts. The values that divide each part are called the first, second, and third quartiles; and they are denoted by Q_1, Q_2, and Q_3, respectively.

$$IQR = Q_3 - Q_1 \tag{130}$$

In other words, the IQR is a measure for the size of the subspace that contains approximately 50% of the data located around the median. Therefore, the IQR can be understood as a measure of statistical dispersion.

As the Figures 18 to 21 suggest, the attenuation coefficient α generally shows a clear variation in its absolute value. Depending on the specific case, also the means and standard deviations are noticeably different. To take account of every single detail, each frequency case will be discussed separately below.

Case 1: $ka = 0.25$

The first case with $ka = 0.25$ is the one with the lowest excitation frequency and therefore with the largest wavelength. A relatively large wavelength compared to the precipitate size ($\lambda/a = 25.13$) results in the smallest mean attenuation coefficient α of approximately $\bar{\alpha}_1 = 1.143m^{-1}$. The relatively small precipitate size causes only a very weak scattering effect and therefore a weak energy deviation and a low-value attenuation coefficient. However, the related Figure 18 shows a very nice and almost symmetric distribution of the measured values.

Case 2: $ka = 0.5$

An increased normalized frequency of $ka = 0.5$ is associated with a decrease in wavelength and therefore smaller wavelength-precipitate size ratio. The smaller ratio in turn leads to a stronger scattering effect and to an increased average attenuation coefficient $\bar{\alpha}_2 = 1.382m^{-1}$. The absolute spread of the data points in this case is slightly

higher than for the first case. Thus the bin size is slightly increased, as 129 suggests. Again, the data shows a Gaussian like distribution around $\bar{\alpha}_2$.

Case 3: $ka = 0.75$

Despite the outlier around $\alpha = 2.03$, the data is very nicely distributed around $\bar{\alpha}_2 = 1.607m^{-1}$.

Case 4: $ka = 1.0$

The data variation for case four can be seen in Figure 21. This case is the one with the most wave interaction with the precipitates. As the wavelength-precipitate size ratio dropped to $\lambda/a = 6.28$ the wavelength is in a region where the same precipitate size as before has the most impact on the propagating wave. Consequently, the attenuation coefficient further grows to an average value of $\bar{\alpha}_2 = 1.815m^{-1}$. The more impactful scattering behavior results in a higher uncertainty in the α distribution and a relatively big range. Therefore, the shape of the illustrated distribution looks subtly odd.

The discussed figures have all one thing in common: they show a distribution of the numerically determined attenuation coefficients. That means the arrangement of the precipitates, with the number density and all other material properties held constant, does have an influence on the material attenuation. Thus, the answer to the question if the spatial distribution of the precipitates is correlated with the attenuation behavior or the attenuation coefficient, respectively, is yes. With that the existence of the correlation is shown, the quantification of that correlation is the second problem and is covered in the following section.

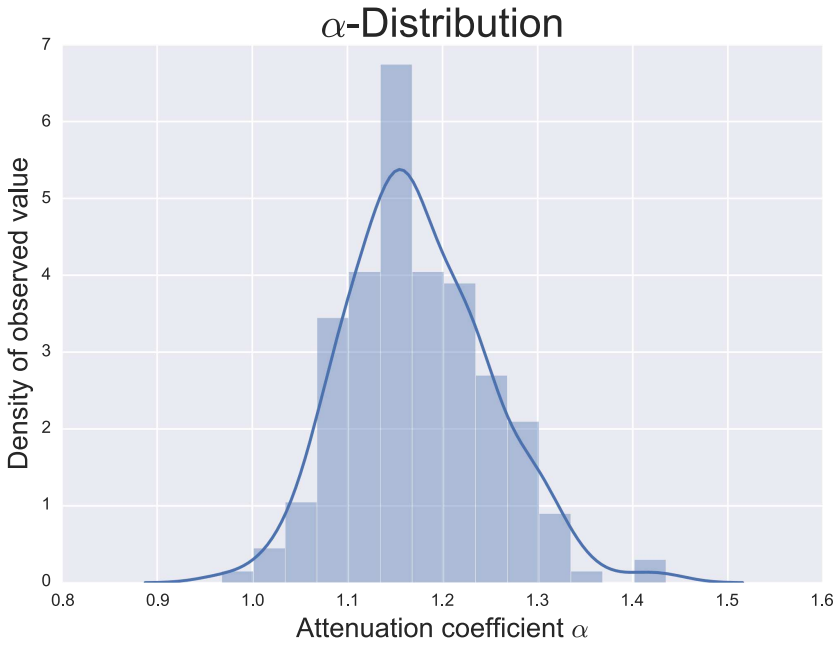


Figure 18: Variation of α for $ka = 0.25$ and 200 data points

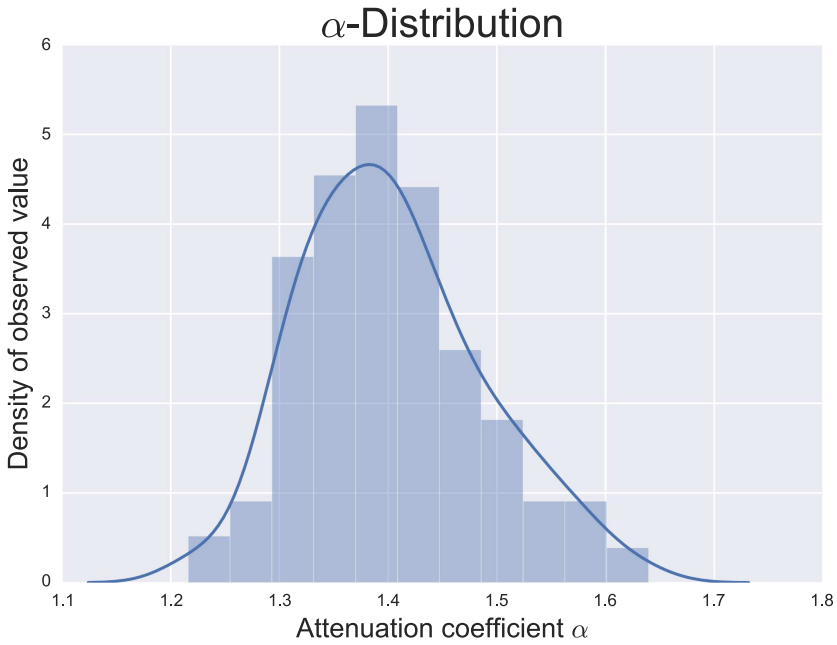


Figure 19: Variation of α for $ka = 0.5$ and 200 data points

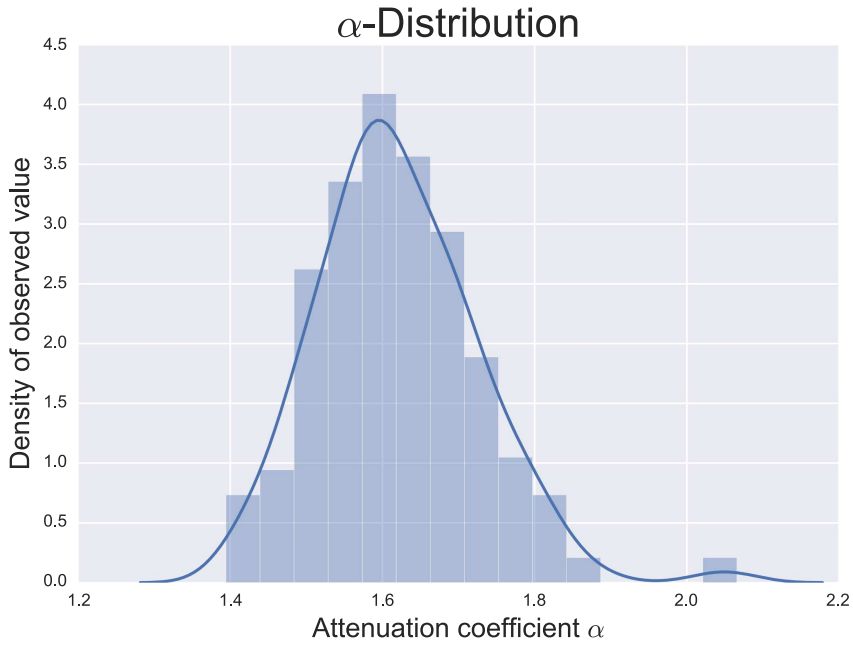


Figure 20: Variation of α for $ka = 0.75$ and 200 data points

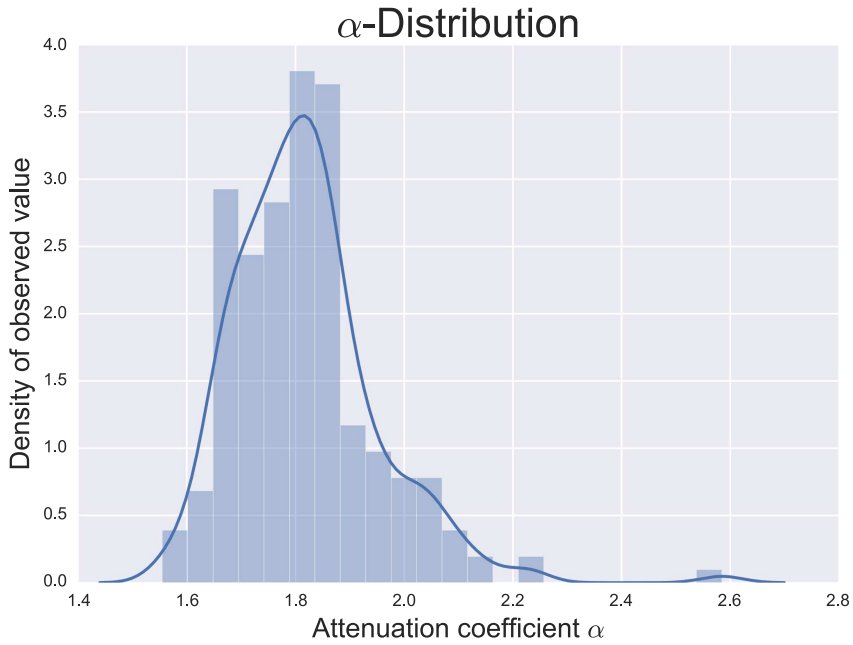


Figure 21: Variation of α for $ka = 1.0$ and 200 data points

5.2 Correlation of Data

This section is concerned with the correlation of the data or to be exact with the qualification of the data correlation. As described in the previous chapter, the correlation between the spatial distribution of the precipitates and the attenuation coefficient α can be assessed by combining both information for each microstructure and perform a (second) principal component analysis. That principal component analysis reduces the data dimension to the size of two principal components which are plotted into a two-dimensional coordinate system, as shown in Figures 22, 23, 24, and 25. Generally, the distribution of the points has a larger spread along the first principal axis compared to the spread along the second and all other axes. To assess the strength of correlation, those spreads have to be weighed up. A relatively big range of the data point locations along the first principal axis compared to the second and all other axes can indicate a correlation or a certain relationship between the input data. On the other hand, similar ranges may imply low or no correlation of the data. It is important to note that the ranges ratio (first and second principal axes) rr_{12}

$$rr_{12} = \frac{PC_{1,max} - PC_{1,min}}{PC_{2,max} - PC_{2,min}} \tag{131}$$

measures the strength of the linear relationship between the two input data sets, but a high value of $|rr_{12}|$ does not necessarily imply a cause and effect relationship or that the two variables are linearly related. It is easy to devise nonlinear relationships that give a high correlation coefficient and it is important to look at the data and use common sense.

The results for the present study are shown in Figures 22 to 25 for the normalized frequency cases $ka = 0.25, 0.5, 0.75$, and 1.0. To analyze each case, the spreads along the first and second principal axes, as well as the ratio of those ranges are listed in Table 5. The ratio rr_{12} and consequently the strength of correlation increase from 4 for the first case to 60 for the fourth case, what strengthens two main assumptions.

Table 5: Overview of principal component ranges and their ratios

Case	**1**	**2**	**3**	**4**
Normalized frequency ka	0.25	0.5	0.75	1.0
Range of first PC	0.3	0.4	0.4	0.6
Range of second PC	0.075	0.02	0.02	0.01
Ranges ratio rr_{12}	4	20	20	60

First, the growing ratio rr_{12} shows the growing influence of the scattering effect for higher normalized frequencies ka. Since ka is the only parameter that changes pointedly throughout this study, it has to be in charge of the increasing ratio. As an increasing ratio requires more information, especially more distinguishable information, an increasing scattering influence and thus a more distinct wave field stands to reason. Second, the principal component analysis is capable of measuring the correlation of the data. Assume the first premise described before is correct and the increasing scattering effect makes the microstructures easier to distinguish between different acoustic properties, the principal component analysis is indeed capable of measuring the changing degree of correlation.

Now, focus on the actual correlation of the depicted data. With a ranges ratio of $rr_{12} = 4$ the first frequency case in Figure 22 shows only a very weak correlation between the spatial distribution of the precipitates and the attenuation coefficient α. The reason for that is most likely the very weak scattering effect that does not generate remarkable scattering and attenuation patterns.

By raising the frequency to $ka = 0.5$ or $ka = 0.75$, the data points tend to be located closer to the first principal axis and therefore the range ratio jumps to $rr_{12} = 20$ for both cases (see Table 5). Higher frequencies lead to smaller wavelengths and therefore to more interaction between the wave and the precipitates whose sizes do not change throughout the present study.

The fourth and last case with the highest normalized frequency $ka = 1.0$ and thus the smallest wavelength is supposed to reveal the most impactful wave-precipitate

interaction. That can be supported by a tripling of the range ratio to a value of $rr_{12} = 60$. Furthermore, the correlation that is detected by the principal component analysis turned to a much stronger relation compared to the first case of $ka = 0.25$ and the related $rr_{12} = 4$. Although Figure 25 shows some exceptional values or outliers, respectively, outside of the main body, the main data bulk is well behaving and can be definitely used for evaluation purposes.

Even though the principal component analysis for the fourth case shows a comparatively strong correlation, the overall performance is mostly weak. Microstructures with a large wavelength to precipitate size ratio do not show a clear correlation between their spatial distribution of precipitates and the attenuation behavior due to a very weak scattering effect. In summary it can be said, that a correlation definitely exists between microstructural and acoustic properties, but the qualification of that relation is fairly difficult. Restricting the statement to high scattering influence, a moderate correlation can be testified.

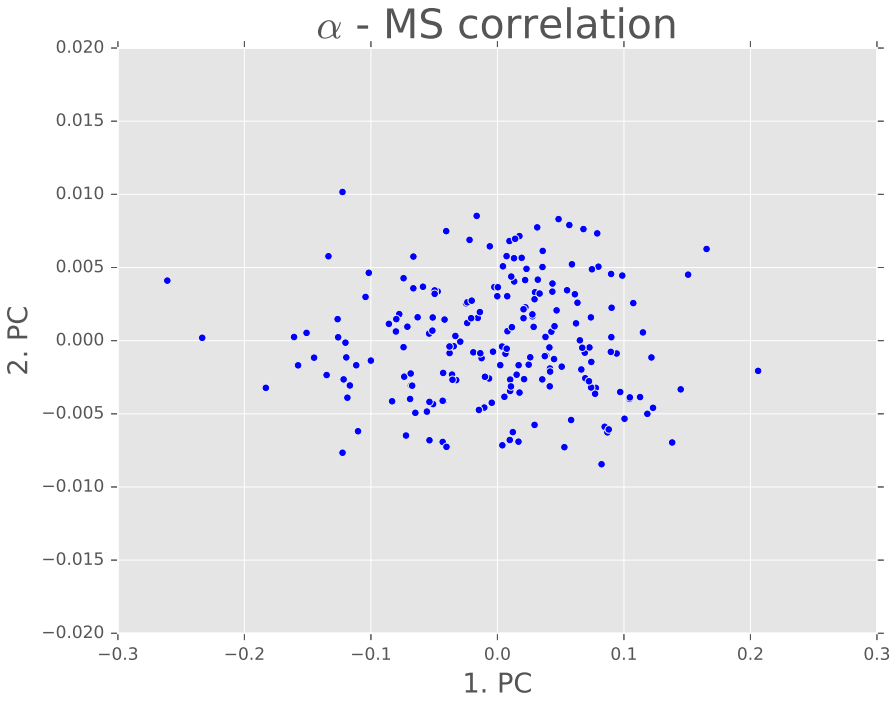


Figure 22: PCA of linked data for $ka = 0.25$

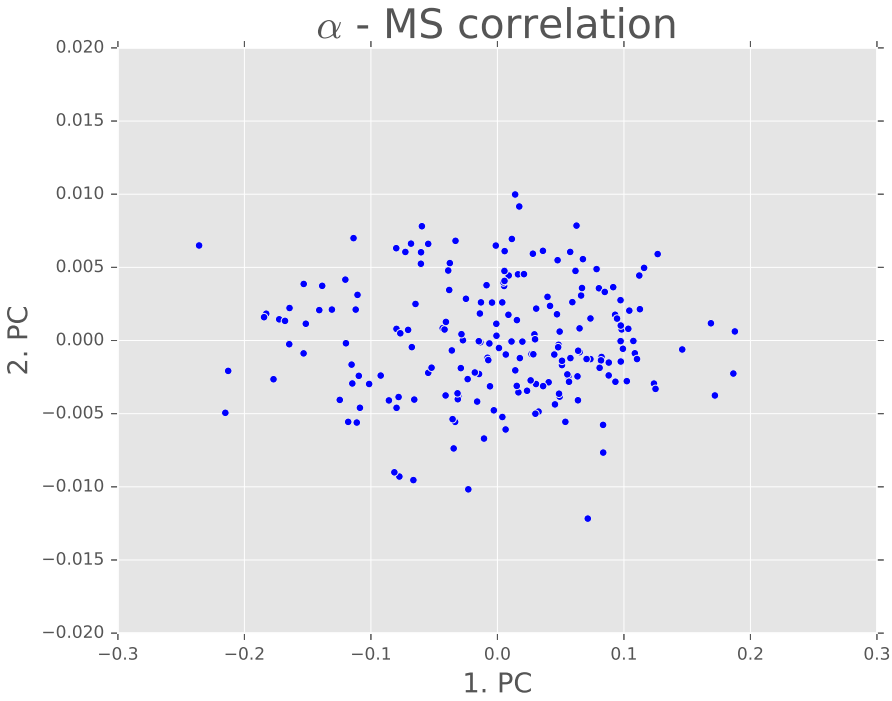


Figure 23: PCA of linked data for $ka = 0.5$

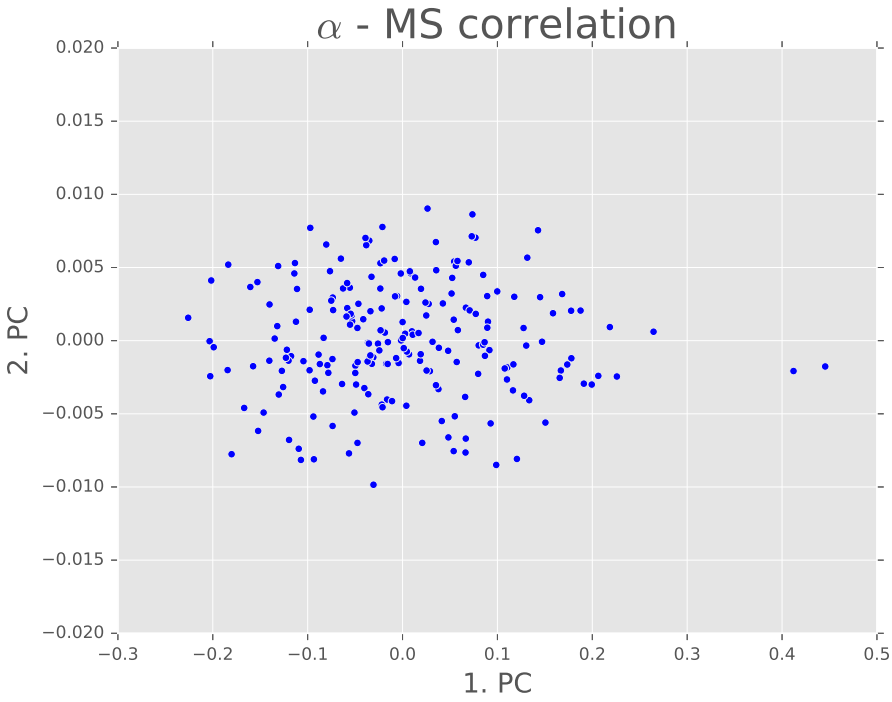


Figure 24: PCA of linked data for $ka = 0.75$

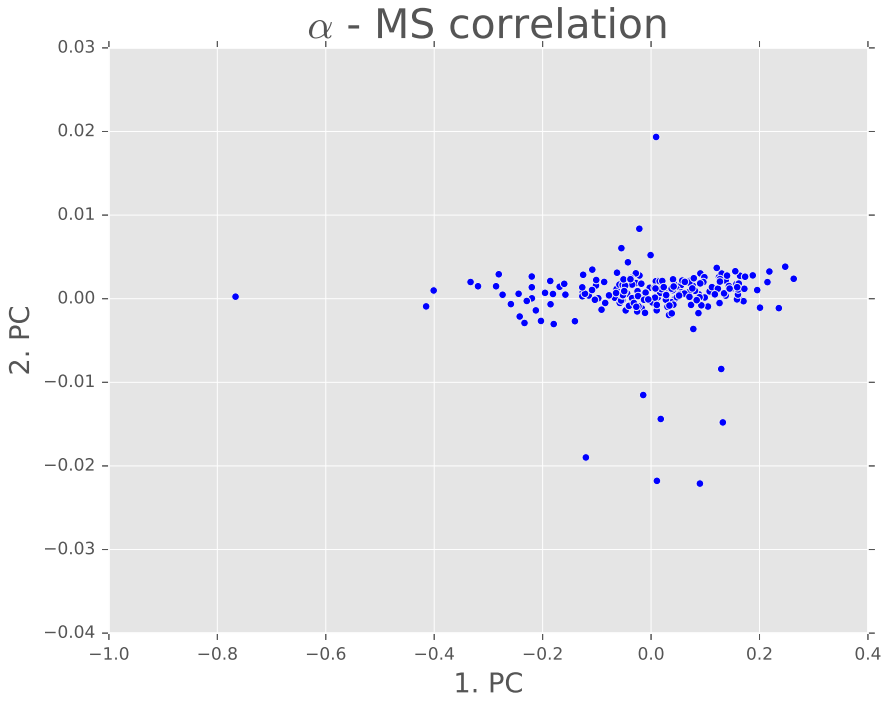


Figure 25: PCA of linked data for $ka = 1.0$

CHAPTER VI

CONCLUSION AND OUTLOOK

This research demonstrates a statistical analysis using statistical and numerical methods to assess the correlation between the spatial distribution of precipitates and the acoustic attenuation behavior of a monocrystalline iron-copper FeCu-steel. Both statistical and numerical methods are described and the existence as well as the qualification of the correlation are shown.

To link microstructural information and acoustic properties, two-dimensional monocrystalline microstructures were virtually generated with a random spatial distribution of precipitates but defined and constant material properties number density, stiffness, and analyzed domain size. The generated microstructures are statistically evaluated with two-point statistics and the principal component analysis. Each microstructure is then discretized and the wave equation is solved on that discretized domain to determine the attenuation behavior of each distinct structure. Lastly, both sides, the spatial distribution information and the attenuation behavior, are combined and correlations are investigated by analyzing the data spread and the first two components of a second principal component analysis of the combined data.

It could be shown that a correlation between the spatial distribution of the precipitates and the attenuation behavior exists by varying the spatial distribution and measuring the prevailing attenuation coefficient α. The attenuation coefficients also showed a certain spread of their values, so that a sensibility of α to the spatial distribution is highly certain. Furthermore, the quality of correlation is investigated. The principal component analysis of the combined microstructural information and acoustic attenuation behavior reveals low range ratios and thus a weak correlation

of the input data for low normalized frequency domains. Higher frequency domains, where the wavelength to precipitate size ratio is smaller and therefore the scattering effect more impactful, the correlation is stronger but still on a moderate level. As the scattering effect plays a key role to correlate microstructural information of the spatial distribution of the precipitates and the related attenuation coefficient, the introduced and applied methods are the most effective for relatively high frequency spectra with a low wavelength to defect size ratio.

The same statistical analysis can and will be applied in future works to determine the influence of precipitate size and stiffness on the attenuation behavior. Moreover, the main focus will be on introducing a nonlinear residual stress field around the precipitates to also investigate the influence of spatial distribution, size, and stiffness of the precipitates on the acoustic nonlinearity parameter β. These studies can potentially contribute to solve the so-called inverse problem and lead to major progress in practical nondestructive evaluation applications.

APPENDIX A

PYTHON CODE

```
#############################################################################
############################## EDITOR'S INFORMATION ##########################
#############################################################################

# Daniel Gruen
# dgruen6@gatech.edu
# Georgia Institute of Technology, Atlanta, GA, USA
# University of Stuttgart, Stuttgart, Germany
#
# This code is part of Daniel Gruen's Master Thesis in the area of Non-Destructive-
# Evaluation (NDE)
#
# Advisors: Dr. Laurence Jacobs, Georgia Institute of Technology
#           Dr. Laurent Capolungo, Los Alamos National Laboratory
#                                  Dr. Jin-Yeon Kim, Georgia Institute of Technology
#
# Date of creation: Spring & Summer 2016



#############################################################################
############################## FUNCTION INFORMATION #########################
#############################################################################
#
# mainfile
#
# This is the control function of the preprocessing and main calculation.
# It has the highest position in the command structure.



#############################################################################
########################## SET MICROSTRUCTURE PROPERTIES ####################
#############################################################################

class structtype():
    pass
microstructure = structtype()

microstructure.stat_analysis = 1 # statistical analysis switch (2point stats and PCA for multi
    analysis)  0: OFF, 1: ON
microstructure.analtype = 1 # 1: single (one microstructure generated), 2: multi (several
    microstructures generated, number defined in <microstructure.no_microstructures>)
microstructure.output = 1 # 0: all outputs suppressed, very quick program flow, necessary for big
    data computing!
```

```
microstructure.location_variation = 1 # switch for variation of precipitate arrangement, 0: OFF,
    1: ON (if OFF standard arrangement is used -> checkerboard)
microstructure.size_variation = 0 # switch for variation of precipitate size, 0: OFF, 1: ON
microstructure.stiffness_variation = 0 # switch for variation of precipitate stiffness, 0: OFF, 1:
    ON

# Enter material properties (edge length of domain is determined automatically!)
microstructure.volume_fraction = 0.0088 # volume fraction (unitless, not percent!! -> min:0 max:
    1)
microstructure.no_precipitates = 200 # number of precipitates in squared domain for statistical
    analysis (FEM analysis might have another number of precipitates!)
microstructure.edgelength_noprecipitates = 20e-8 # edge length if zero precipitates selected
microstructure.precipitate_radius = 2e-9 # radius of precipitate in m
microstructure.min_preccenter_distance = 4.00001e-9 # minimum distance between two centers of
    precipitates (in nm)
microstructure.size_stdev = 0.15 # standard deviation of of precipitate size in percent
microstructure.stiffness_stdev = 0.15 # standard deviation of of precipitate stiffness in percent
microstructure.voxpedge = 600 # number of voxels along edge (squared domain)
microstructure.no_microstructures = 5 # number of randomly generated microstructures
microstructure.check_microstructure = 0 # Option to plot microstructure when several structures
    are created (see description below)
# (0: OFF, 1: ON, choose # of MS below, 2: ON, MS with max and min principal values plotted), 3:
    ON, 1. PC with max, zero and min PV while all other PV of other PC nearly zero
microstructure.check_microstructure_no = 0 # number of microstructure to be plotted (for option
    microstructure.check_microstructure = 1)
microstructure.no_components_variance_plot = 2 # number of components being plotted in the
    component variance plot


###############################################################################
############################## SET FEM PROPERTIES ##############################
###############################################################################

fem = structtype()

fem.analysis = 0 # main switch for FEM analysis (0: OFF, 1: ON)

fem.convergence_analysis = 0 # switch for convergence analysis (0: OFF, 1: ON)
fem.convergence_analtype = 1.5 # 1: stop when error drops below threshold, 2: stop after defined
    number of steps
fem.convergence_threshold = 0.05 # minimum relative difference of mean errors between two steps
    that indicates convergence
fem.convergence_nosteps = 1 # number of steps in convergence analysis (at least 2)
fem.convergence_edge_factor = 1.5 # increasing factor of elements elong edge in convergence
    analysis
fem.convergence_precipitate_factor = 1.5 # increasing factor of elements elong precipitate
    boundary in convergence analysis
fem.microstructure_output = 1 # save microstructure as vtk/eps image (0: OFF, 1: ON) (IF ON,
    convergence analysis very slow since user has to close window every loop)
fem.save_animation = 1 # save animated wave as vtk-file? (0: NO, 1:YES) (turn off for HPC usage!)

fem.dimension = 2 # dimension of wave equation (1: 1D, 2:2D)
fem.solver = 'CG' # FEM solver (CG, sparsesolver,...)
fem.elements_factor_precipitate_boundary = 3 # factor by which elements on precipitate boundary
    are smaller than on the edges (respect different wave velocity in prec.)
```

```
fem.no_measurement_lines = 10 # number of evenly spaced measurement lines along which the mean
    displacement is determined (lines are orthogonal to propagation direction) (number of points
    on each line correlates with element size)
fem.no_measurement_lines_prop = 100 # number of evenly spaced measurement lines along the
    propagation direction (lines are parallel to propagation direction) (number of points on each
    line correlates with element size)
fem.no_point_per_line_factor_prop = 10 # factor by that the number of measurement points on the
    propagation line is higher than the natural number of points (element size)
fem.time_integration = 2 # time integration scheme (1: explicit, 2:implicit with bathe-method)
fem.element_type = 1 # element type (1:linear, 2:quadratic)
fem.elements_per_wavelength = 10 # elements along excitation edge (start value for convergence
    analysis) [number of elements along other edge are set automatically]
fem.autotime = 1 # automatic time step and endtime determination by using CFL number and only one
    edgelength propagation (0: OFF, 1:ON)
fem.timestep = 4.25e-13 # time step in seconds (start value for convergence analysis)
fem.simulation_starttime = 0.0 # start time for which equation is solved
fem.simulation_endtime = 0.000000000028137757 #4.4e-9 # end time for which equation is solved
fem.cfl = 1.0 # CFL number (if fem.autotime == 1)
fem.no_excwaves = 100 # number of waves in excitation wavepacket

fem.excitation_frequency = 1.17e11 #50e9 # excitation frequency in Hz
fem.excitation_amplitude = 1e-6 # excitation amplitude in m
fem.precipitate_pwavevel = 4760 # p-wave velocity in precipitate in m/s (copper)
fem.precipitate_swavevel = 2325 # s-wave velocity in precipitate in m/s (copper)
fem.precipitate_rho = 8960 # precipitate density in kg/m^3
fem.lattice_pwavevel = 5912 # p-wave velocity in lattice structure in m/s (iron)
fem.lattice_swavevel = 3240 # s-wave velocity in lattice structure in m/s (iron)
fem.lattice_rho = 7874 # lattice density in kg/m^3



################################################################################
################### CHECK DEFINED OPTIONS AND PRINT THEM ########################
################################################################################

import sys
import os
sys.path.append('.' + os.sep + 'Subfunctions')
from function_set1 import *

check_options(microstructure,fem)


################################################################################
########################### PATHNAMES AND FILENAMES ############################
################################################################################

import time

# Define pathnames. Relative pathnames!
pathnames = structtype()

currenttime = time.strftime('Date_%Y_%m_%d_Time_%H_%M_%S')

pathnames.all_results = '.' + os.sep +'Results' + os.sep + currenttime
```

```
pathnames.all_results_no_dot = pathnames.all_results[1:]
pathnames.statistical_results = pathnames.all_results + os.sep + 'Statistical_analysis'
pathnames.fem_results = pathnames.all_results + os.sep + 'FreeFEM_results'
pathnames.fem_results_globalvariables = pathnames.all_results + os.sep + 'Python_Data'
pathnames.fem_results_each_ms = pathnames.fem_results + os.sep + 'Microstructure_'
pathnames.fem_convergence = pathnames.all_results + os.sep + 'FEM_convergence_analysis'
pathnames.mainfile_folderpath = os.path.dirname(os.path.realpath(__file__))

# Define filenames.
filenames = structtype()

filenames.infofile = 'Defined_properties.txt'
filenames.pca_plot = 'PCA_plot'
filenames.fem_input = 'FreeFEM_input.edp'
filenames.fem_results = 'Wave_out.txt'
filenames.fem_meshinfofile = 'Meshinfo.txt'
filenames.fem_meshimage = 'Meshimage.eps'
filenames.convergence_displacements = 'Convergence_displacements.eps'
filenames.convergence_error = 'Convergence_error.eps'
filenames.globalsave = currenttime + '_data_'
filenames.fem_displacements = 'Displacements.txt'

###############################################################################
############################ CREATE FOLDER STRUCTURE ##########################
###############################################################################

from microstructure_subfunctions import *
from fem_subfunctions import *

# Create folders
os.makedirs(pathnames.all_results)
os.makedirs(pathnames.statistical_results)
os.makedirs(pathnames.fem_results)
os.makedirs(pathnames.fem_results_globalvariables)
if fem.convergence_analysis == 1:
    os.makedirs(pathnames.fem_convergence)
if microstructure.analtype == 2:
    for ms in range(1,microstructure.no_microstructures+1):
        os.makedirs(pathnames.fem_results_each_ms+str(ms))
else:
    os.makedirs(pathnames.fem_results_each_ms+'1')


###############################################################################
########################### STATISTICAL ANALYSIS ##############################
###############################################################################

(microstructure, statistical_results) = statistical_analysis(microstructure,fem,pathnames,
    filenames)

# statistical_results:
#                       .loc_precipitates (3D)
#                       .vox_matrix       (3D)
#                       .lattice_point    (2D)
#                       .principal_components (2D)
```

```
##############################################################################
############################### FEM CALCULATION ##############################
##############################################################################

fem_results = structtype()
statistical_results_2save = structtype()

# read FreeFEM results (.txt) and save FreeFEM++ results as python file
fem_results = fem_analysis(microstructure, fem, pathnames, filenames, statistical_results,
    fem_results, statistical_results_2save)

# fem_results:
#                .displacements (2D: simulation time, x-loc, y-loc, y-displacement)
```

APPENDIX B

FREEFEM++ CODE

```
// Location variation, 200 precipitates

verbosity=0;
load "msh3"
load "iovtk" //load library of vtk

int C1=100;
int C2=200;
int C3=300;
int C4=400;
int C5=500;
int C6=600;
int C7=700;
int C8=800;
int C9=900;
int C10=1000;
int C11=1100;
int C12=1200;
int C13=1300;
int C14=1400;
int C15=1500;
int C16=1600;
int C17=1700;
int C18=1800;
int C19=1900;
int C20=2000;
int C21=2100;
int C22=2200;
int C23=2300;
int C24=2400;
int C25=2500;
int C26=2600;
int C27=2700;
int C28=2800;
int C29=2900;
int C30=3000;
int C31=3100;
int C32=3200;
int C33=3300;
int C34=3400;
int C35=3500;
int C36=3600;
int C37=3700;
int C38=3800;
int C39=3900;
int C40=4000;
```

```
int C41=4100;
int C42=4200;
int C43=4300;
int C44=4400;
int C45=4500;
int C46=4600;
int C47=4700;
int C48=4800;
int C49=4900;
int C50=5000;
int C51=5100;
int C52=5200;
int C53=5300;
int C54=5400;

//Building the Mesh

border a0(t=0,0.000000225189) { x=0.000000225189; y= t ; label=C1;}
border a1(t=0,0.000000225189) { x=0.000000225189-t; y=0.000000225189; label=C2;}
border a2(t=0,0.000000225189) { x=0; y= 0.000000225189-t ; label=C3;}
border a3(t=0,0.000000225189) { x=t; y=0 ;label=C4;}

border b1(t=0,2*pi){x=0.000000041907+0.000000002000*cos(t); y=0.000000097382+0.000000002000*sin(t)
    ; label=C5;}
border b2(t=0,2*pi){x=0.000000163743+0.000000002000*cos(t); y=0.000000219099+0.000000002000*sin(t)
    ; label=C6;}
border b3(t=0,2*pi){x=0.000000026799+0.000000002000*cos(t); y=0.000000015036+0.000000002000*sin(t)
    ; label=C7;}
border b4(t=0,2*pi){x=0.000000067295+0.000000002000*cos(t); y=0.000000092266+0.000000002000*sin(t)
    ; label=C8;}
border b5(t=0,2*pi){x=0.000000221685+0.000000002000*cos(t); y=0.000000063299+0.000000002000*sin(t)
    ; label=C9;}
border b6(t=0,2*pi){x=0.000000025079+0.000000002000*cos(t); y=0.000000121478+0.000000002000*sin(t)
    ; label=C10;}
border b7(t=0,2*pi){x=0.000000213151+0.000000002000*cos(t); y=0.000000113524+0.000000002000*sin(t)
    ; label=C11;}
border b8(t=0,2*pi){x=0.000000209479+0.000000002000*cos(t); y=0.000000042031+0.000000002000*sin(t)
    ; label=C12;}
border b9(t=0,2*pi){x=0.000000033220+0.000000002000*cos(t); y=0.000000071655+0.000000002000*sin(t)
    ; label=C13;}
border b10(t=0,2*pi){x=0.000000117830+0.000000002000*cos(t); y=0.000000161841+0.000000002000*sin(t
    ); label=C14;}
border b11(t=0,2*pi){x=0.000000209029+0.000000002000*cos(t); y=0.000000173194+0.000000002000*sin(t
    ); label=C15;}
border b12(t=0,2*pi){x=0.000000069027+0.000000002000*cos(t); y=0.000000033109+0.000000002000*sin(t
    ); label=C16;}
border b13(t=0,2*pi){x=0.000000075445+0.000000002000*cos(t); y=0.000000188659+0.000000002000*sin(t
    ); label=C17;}
border b14(t=0,2*pi){x=0.000000131376+0.000000002000*cos(t); y=0.000000155621+0.000000002000*sin(t
    ); label=C18;}
border b15(t=0,2*pi){x=0.000000193538+0.000000002000*cos(t); y=0.000000092872+0.000000002000*sin(t
    ); label=C19;}
border b16(t=0,2*pi){x=0.000000094845+0.000000002000*cos(t); y=0.000000052308+0.000000002000*sin(t
    ); label=C20;}
```

```
border b17(t=0,2*pi){x=0.000000178375+0.000000002000*cos(t); y=0.000000176374+0.000000002000*sin(t
    ); label=C21;}
border b18(t=0,2*pi){x=0.000000206139+0.000000002000*cos(t); y=0.000000032343+0.000000002000*sin(t
    ); label=C22;}
border b19(t=0,2*pi){x=0.000000111823+0.000000002000*cos(t); y=0.000000083824+0.000000002000*sin(t
    ); label=C23;}
border b20(t=0,2*pi){x=0.000000109762+0.000000002000*cos(t); y=0.000000007408+0.000000002000*sin(t
    ); label=C24;}
border b21(t=0,2*pi){x=0.000000200882+0.000000002000*cos(t); y=0.000000009868+0.000000002000*sin(t
    ); label=C25;}
border b22(t=0,2*pi){x=0.000000050097+0.000000002000*cos(t); y=0.000000063690+0.000000002000*sin(t
    ); label=C26;}
border b23(t=0,2*pi){x=0.000000194882+0.000000002000*cos(t); y=0.000000176595+0.000000002000*sin(t
    ); label=C27;}
border b24(t=0,2*pi){x=0.000000113312+0.000000002000*cos(t); y=0.000000196076+0.000000002000*sin(t
    ); label=C28;}
border b25(t=0,2*pi){x=0.000000175133+0.000000002000*cos(t); y=0.000000218465+0.000000002000*sin(t
    ); label=C29;}
border b26(t=0,2*pi){x=0.000000159398+0.000000002000*cos(t); y=0.000000204260+0.000000002000*sin(t
    ); label=C30;}
border b27(t=0,2*pi){x=0.000000091418+0.000000002000*cos(t); y=0.000000081900+0.000000002000*sin(t
    ); label=C31;}
border b28(t=0,2*pi){x=0.000000015278+0.000000002000*cos(t); y=0.000000072338+0.000000002000*sin(t
    ); label=C32;}
border b29(t=0,2*pi){x=0.000000214311+0.000000002000*cos(t); y=0.000000131690+0.000000002000*sin(t
    ); label=C33;}
border b30(t=0,2*pi){x=0.000000028950+0.000000002000*cos(t); y=0.000000007548+0.000000002000*sin(t
    ); label=C34;}
border b31(t=0,2*pi){x=0.000000023045+0.000000002000*cos(t); y=0.000000176029+0.000000002000*sin(t
    ); label=C35;}
border b32(t=0,2*pi){x=0.000000200612+0.000000002000*cos(t); y=0.000000106518+0.000000002000*sin(t
    ); label=C36;}
border b33(t=0,2*pi){x=0.000000159004+0.000000002000*cos(t); y=0.000000141889+0.000000002000*sin(t
    ); label=C37;}
border b34(t=0,2*pi){x=0.000000025994+0.000000002000*cos(t); y=0.000000138290+0.000000002000*sin(t
    ); label=C38;}
border b35(t=0,2*pi){x=0.000000094399+0.000000002000*cos(t); y=0.000000215263+0.000000002000*sin(t
    ); label=C39;}
border b36(t=0,2*pi){x=0.000000084135+0.000000002000*cos(t); y=0.000000220913+0.000000002000*sin(t
    ); label=C40;}
border b37(t=0,2*pi){x=0.000000026415+0.000000002000*cos(t); y=0.000000168884+0.000000002000*sin(t
    ); label=C41;}
border b38(t=0,2*pi){x=0.000000084513+0.000000002000*cos(t); y=0.000000175444+0.000000002000*sin(t
    ); label=C42;}
border b39(t=0,2*pi){x=0.000000045533+0.000000002000*cos(t); y=0.000000134811+0.000000002000*sin(t
    ); label=C43;}
border b40(t=0,2*pi){x=0.000000041026+0.000000002000*cos(t); y=0.000000005111+0.000000002000*sin(t
    ); label=C44;}
border b41(t=0,2*pi){x=0.000000140919+0.000000002000*cos(t); y=0.000000217355+0.000000002000*sin(t
    ); label=C45;}
border b42(t=0,2*pi){x=0.000000121876+0.000000002000*cos(t); y=0.000000214829+0.000000002000*sin(t
    ); label=C46;}
border b43(t=0,2*pi){x=0.000000095291+0.000000002000*cos(t); y=0.000000189185+0.000000002000*sin(t
    ); label=C47;}
```

```
border b44(t=0,2*pi){x=0.000000116923+0.000000002000*cos(t); y=0.000000111145+0.000000002000*sin(t
    ); label=C48;}
border b45(t=0,2*pi){x=0.000000028845+0.000000002000*cos(t); y=0.000000129697+0.000000002000*sin(t
    ); label=C49;}
border b46(t=0,2*pi){x=0.000000181037+0.000000002000*cos(t); y=0.000000112767+0.000000002000*sin(t
    ); label=C50;}
border b47(t=0,2*pi){x=0.000000168414+0.000000002000*cos(t); y=0.000000089493+0.000000002000*sin(t
    ); label=C51;}
border b48(t=0,2*pi){x=0.000000117945+0.000000002000*cos(t); y=0.000000193586+0.000000002000*sin(t
    ); label=C52;}
border b49(t=0,2*pi){x=0.000000210268+0.000000002000*cos(t); y=0.000000160765+0.000000002000*sin(t
    ); label=C53;}
border b50(t=0,2*pi){x=0.000000199800+0.000000002000*cos(t); y=0.000000137039+0.000000002000*sin(t
    ); label=C54;}

real dt=0.000000000000854701,Ts=0.000000000000000000,Tf=0.000000000038090164,tcurrent,tmiddle;
real g=0.;
real freq=117000000000.000000000000;
real ampl=0.000001000000;
func perio = [[100,y],[300,y]];

mesh Th=buildmesh(a0(44)+a1(44)+a2(44)+a3(44)+b1(7)+b2(7)+b3(7)+b4(7)+b5(7)+b6(7)+b7(7)+b8(7)+b9
    (7)+b10(7)+b11(7)+b12(7)+b13(7)+b14(7)+b15(7)+b16(7)+b17(7)+b18(7)+b19(7)+b20(7)+b21(7)+b22(7)
    +b23(7)+b24(7)+b25(7)+b26(7)+b27(7)+b28(7)+b29(7)+b30(7)+b31(7)+b32(7)+b33(7)+b34(7)+b35(7)+
    b36(7)+b37(7)+b38(7)+b39(7)+b40(7)+b41(7)+b42(7)+b43(7)+b44(7)+b45(7)+b46(7)+b47(7)+b48(7)+b49
    (7)+b50(7));
fespace Vh(Th,P1,periodic=perio);
fespace Wh(Th,P0);

int inclusion1 = Th(0.000000041907,0.000000097382).region;
int inclusion2 = Th(0.000000163743,0.000000219099).region;
int inclusion3 = Th(0.000000026799,0.000000015036).region;
int inclusion4 = Th(0.000000067295,0.000000092266).region;
int inclusion5 = Th(0.000000221685,0.000000063299).region;
int inclusion6 = Th(0.000000025079,0.000000121478).region;
int inclusion7 = Th(0.000000213151,0.000000113524).region;
int inclusion8 = Th(0.000000209479,0.000000042031).region;
int inclusion9 = Th(0.000000033220,0.000000071655).region;
int inclusion10 = Th(0.000000117830,0.000000161841).region;
int inclusion11 = Th(0.000000209029,0.000000173194).region;
int inclusion12 = Th(0.000000069027,0.000000033109).region;
int inclusion13 = Th(0.000000075445,0.000000188659).region;
int inclusion14 = Th(0.000000131376,0.000000155621).region;
int inclusion15 = Th(0.000000193538,0.000000092872).region;
int inclusion16 = Th(0.000000094845,0.000000052308).region;
int inclusion17 = Th(0.000000178375,0.000000176374).region;
int inclusion18 = Th(0.000000206139,0.000000032343).region;
int inclusion19 = Th(0.000000111823,0.000000083824).region;
int inclusion20 = Th(0.000000109762,0.000000007408).region;
int inclusion21 = Th(0.000000200882,0.000000009868).region;
int inclusion22 = Th(0.000000050097,0.000000063690).region;
int inclusion23 = Th(0.000000194882,0.000000176595).region;
int inclusion24 = Th(0.000000113312,0.000000196076).region;
int inclusion25 = Th(0.000000175133,0.000000218465).region;
int inclusion26 = Th(0.000000159398,0.000000204260).region;
```

```
int inclusion27 = Th(0.000000091418,0.000000081900).region;
int inclusion28 = Th(0.000000015278,0.000000072338).region;
int inclusion29 = Th(0.000000214311,0.000000131690).region;
int inclusion30 = Th(0.000000028950,0.000000007548).region;
int inclusion31 = Th(0.000000023045,0.000000176029).region;
int inclusion32 = Th(0.000000200612,0.000000106518).region;
int inclusion33 = Th(0.000000159004,0.000000141889).region;
int inclusion34 = Th(0.000000025994,0.000000138290).region;
int inclusion35 = Th(0.000000094399,0.000000215263).region;
int inclusion36 = Th(0.000000084135,0.000000220913).region;
int inclusion37 = Th(0.000000026415,0.000000168884).region;
int inclusion38 = Th(0.000000084513,0.000000175444).region;
int inclusion39 = Th(0.000000045533,0.000000134811).region;
int inclusion40 = Th(0.000000041026,0.000000005111).region;
int inclusion41 = Th(0.000000140919,0.000000217355).region;
int inclusion42 = Th(0.000000121876,0.000000214829).region;
int inclusion43 = Th(0.000000095291,0.000000189185).region;
int inclusion44 = Th(0.000000116923,0.000000111145).region;
int inclusion45 = Th(0.000000028845,0.000000129697).region;
int inclusion46 = Th(0.000000181037,0.000000112767).region;
int inclusion47 = Th(0.000000168414,0.000000089493).region;
int inclusion48 = Th(0.000000117945,0.000000193586).region;
int inclusion49 = Th(0.000000210268,0.000000160765).region;
int inclusion50 = Th(0.000000199800,0.000000137039).region;
int effective=Th(0.000000000188,0.000000000188).region;

// Define measurement lines
int nolines = 10;
int ppline = 44;
real[int] xlocations(ppline);
real[int] ylocations(nolines);
for (int ii=0; ii<ppline; ii++){
    xlocations[ii] = ii*0.000000225189051/(ppline-1);
    }
for (int jj=0; jj<nolines; jj++){
    ylocations[jj] = jj*0.000000225189051/(nolines-1);
    }

ofstream fout("Displacements.txt", append);
fout<< "FILE INFORMATION: DISPLACEMENT AT DISTINCT POINTS IN TIME AND DOMAIN (INTERPOLATED) (x-
    coord, y-coord, y-displacement)" << endl;

real rholattice=7874.000000;
real rhoprecip=8960.000000;
real vplattice=5912.000000;
real vslattice=3240.000000;

real lambdalattice = rholattice*(vplattice^2-2*vslattice^2);
real mulattice = rholattice*vslattice^2;
```

```
Wh lambda=lambdalattice *(region==effective)+-1230886400.000000*(region==inclusion1)
    +-1230886400.000000*(region==inclusion2)+-1230886400.000000*(region==inclusion3)
    +-1230886400.000000*(region==inclusion4)+-1230886400.000000*(region==inclusion5)
    +-1230886400.000000*(region==inclusion6)+-1230886400.000000*(region==inclusion7)
    +-1230886400.000000*(region==inclusion8)+-1230886400.000000*(region==inclusion9)
    +-1230886400.000000*(region==inclusion10)+-1230886400.000000*(region==inclusion11)
    +-1230886400.000000*(region==inclusion12)+-1230886400.000000*(region==inclusion13)
    +-1230886400.000000*(region==inclusion14)+-1230886400.000000*(region==inclusion15)
    +-1230886400.000000*(region==inclusion16)+-1230886400.000000*(region==inclusion17)
    +-1230886400.000000*(region==inclusion18)+-1230886400.000000*(region==inclusion19)
    +-1230886400.000000*(region==inclusion20)+-1230886400.000000*(region==inclusion21)
    +-1230886400.000000*(region==inclusion22)+-1230886400.000000*(region==inclusion23)
    +-1230886400.000000*(region==inclusion24)+-1230886400.000000*(region==inclusion25)
    +-1230886400.000000*(region==inclusion26)+-1230886400.000000*(region==inclusion27)
    +-1230886400.000000*(region==inclusion28)+-1230886400.000000*(region==inclusion29)
    +-1230886400.000000*(region==inclusion30)+-1230886400.000000*(region==inclusion31)
    +-1230886400.000000*(region==inclusion32)+-1230886400.000000*(region==inclusion33)
    +-1230886400.000000*(region==inclusion34)+-1230886400.000000*(region==inclusion35)
    +-1230886400.000000*(region==inclusion36)+-1230886400.000000*(region==inclusion37)
    +-1230886400.000000*(region==inclusion38)+-1230886400.000000*(region==inclusion39)
    +-1230886400.000000*(region==inclusion40)+-1230886400.000000*(region==inclusion41)
    +-1230886400.000000*(region==inclusion42)+-1230886400.000000*(region==inclusion43)
    +-1230886400.000000*(region==inclusion44)+-1230886400.000000*(region==inclusion45)
    +-1230886400.000000*(region==inclusion46)+-1230886400.000000*(region==inclusion47)
    +-1230886400.000000*(region==inclusion48)+-1230886400.000000*(region==inclusion49)
    +-1230886400.000000*(region==inclusion50);
Wh mu=mulattice *(region==effective)+1189759744.000000*(region==inclusion1)+1189759744.000000*(
    region==inclusion2)+1189759744.000000*(region==inclusion3)+1189759744.000000*(region==
    inclusion4)+1189759744.000000*(region==inclusion5)+1189759744.000000*(region==inclusion6)
    +1189759744.000000*(region==inclusion7)+1189759744.000000*(region==inclusion8)
    +1189759744.000000*(region==inclusion9)+1189759744.000000*(region==inclusion10)
    +1189759744.000000*(region==inclusion11)+1189759744.000000*(region==inclusion12)
    +1189759744.000000*(region==inclusion13)+1189759744.000000*(region==inclusion14)
    +1189759744.000000*(region==inclusion15)+1189759744.000000*(region==inclusion16)
    +1189759744.000000*(region==inclusion17)+1189759744.000000*(region==inclusion18)
    +1189759744.000000*(region==inclusion19)+1189759744.000000*(region==inclusion20)
    +1189759744.000000*(region==inclusion21)+1189759744.000000*(region==inclusion22)
    +1189759744.000000*(region==inclusion23)+1189759744.000000*(region==inclusion24)
    +1189759744.000000*(region==inclusion25)+1189759744.000000*(region==inclusion26)
    +1189759744.000000*(region==inclusion27)+1189759744.000000*(region==inclusion28)
    +1189759744.000000*(region==inclusion29)+1189759744.000000*(region==inclusion30)
    +1189759744.000000*(region==inclusion31)+1189759744.000000*(region==inclusion32)
    +1189759744.000000*(region==inclusion33)+1189759744.000000*(region==inclusion34)
    +1189759744.000000*(region==inclusion35)+1189759744.000000*(region==inclusion36)
    +1189759744.000000*(region==inclusion37)+1189759744.000000*(region==inclusion38)
    +1189759744.000000*(region==inclusion39)+1189759744.000000*(region==inclusion40)
    +1189759744.000000*(region==inclusion41)+1189759744.000000*(region==inclusion42)
    +1189759744.000000*(region==inclusion43)+1189759744.000000*(region==inclusion44)
    +1189759744.000000*(region==inclusion45)+1189759744.000000*(region==inclusion46)
    +1189759744.000000*(region==inclusion47)+1189759744.000000*(region==inclusion48)
    +1189759744.000000*(region==inclusion49)+1189759744.000000*(region==inclusion50);
```

```
Wh rho=rholattice *(region==effective)+rhoprecip*(region==inclusion1)+rhoprecip*(region==
    inclusion2)+rhoprecip*(region==inclusion3)+rhoprecip*(region==inclusion4)+rhoprecip*(region==
    inclusion5)+rhoprecip*(region==inclusion6)+rhoprecip*(region==inclusion7)+rhoprecip*(region==
    inclusion8)+rhoprecip*(region==inclusion9)+rhoprecip*(region==inclusion10)+rhoprecip*(region==
    inclusion11)+rhoprecip*(region==inclusion12)+rhoprecip*(region==inclusion13)+rhoprecip*(region
    ==inclusion14)+rhoprecip*(region==inclusion15)+rhoprecip*(region==inclusion16)+rhoprecip*(
    region==inclusion17)+rhoprecip*(region==inclusion18)+rhoprecip*(region==inclusion19)+rhoprecip
    *(region==inclusion20)+rhoprecip*(region==inclusion21)+rhoprecip*(region==inclusion22)+
    rhoprecip*(region==inclusion23)+rhoprecip*(region==inclusion24)+rhoprecip*(region==inclusion25
    )+rhoprecip*(region==inclusion26)+rhoprecip*(region==inclusion27)+rhoprecip*(region==
    inclusion28)+rhoprecip*(region==inclusion29)+rhoprecip*(region==inclusion30)+rhoprecip*(region
    ==inclusion31)+rhoprecip*(region==inclusion32)+rhoprecip*(region==inclusion33)+rhoprecip*(
    region==inclusion34)+rhoprecip*(region==inclusion35)+rhoprecip*(region==inclusion36)+rhoprecip
    *(region==inclusion37)+rhoprecip*(region==inclusion38)+rhoprecip*(region==inclusion39)+
    rhoprecip*(region==inclusion40)+rhoprecip*(region==inclusion41)+rhoprecip*(region==inclusion42
    )+rhoprecip*(region==inclusion43)+rhoprecip*(region==inclusion44)+rhoprecip*(region==
    inclusion45)+rhoprecip*(region==inclusion46)+rhoprecip*(region==inclusion47)+rhoprecip*(region
    ==inclusion48)+rhoprecip*(region==inclusion49)+rhoprecip*(region==inclusion50);

// Define problem

Vh u1h,u2h,u1h0=0.0,u2h0=0.0,u1h1=0.0,u2h1=0.0,u1h2,u2h2,v1h,v2h,v1h2,v2h2,u1h1dot=0.,u1h1ddot=0.,
    u1h2dot,u1hdot,u1hddot,u2h1dot=0.,u2h1ddot=0.,u2h2dot,u2hdot,u2hddot;
macro Grad(u1,u2) [dx(u1),dy(u1),dx(u2),dy(u2)]//
func C = [[lambda+2*mu, 0, 0, lambda],[0, mu, mu, 0],[0, mu, mu, 0],[lambda, 0, 0, lambda+2*mu]];

problem step1([u1h2,u2h2],[v1h2,v2h2],solver=CG) = int2d(Th)(Grad(v1h2,v2h2)'*C*(dt^2)*Grad(u1h2,
    u2h2)*rho^(-1))
-----------------------------------------------------+int2d(Th)(16*u1h2*v1h2)-int2d(Th)(16*u1h1*
    v1h2+8*u1h1dot*v1h2*dt+u1h1ddot*v1h2*dt^2)
-----------------------------------------------------+int2d(Th)(16*u2h2*v2h2)-int2d(Th)(16*u2h1*
    v2h2+8*u2h1dot*v2h2*dt+u2h1ddot*v2h2*dt^2)
-----------------------------------------------------+on(400,u1h2=0.0,u2h2=ampl*sin(2*pi*freq*
    tmiddle));

problem step2([u1h,u2h],[v1h,v2h],solver=CG) = int2d(Th)(Grad(v1h,v2h)'*C*(dt^2)*Grad(u1h,u2h)*rho
    ^(-1))
                                              + int2d(Th)(u1h1dot*v1h*dt - 4*u1h2dot*v1h*dt + 3*
                                                  u1h1*v1h - 12*u1h2*v1h) + int2d(Th)(9*u1h*v1h)
                                              + int2d(Th)(u2h1dot*v2h*dt - 4*u2h2dot*v2h*dt + 3*
                                                  u2h1*v2h - 12*u2h2*v2h) + int2d(Th)(9*u2h*v2h)
                                              + on(400,u1h=0.0,u2h=ampl*sin(2*pi*freq*tcurrent));

func funcu1h2dot = (u1h2-u1h1)*(4/dt)-u1h1dot;
func funcu2h2dot = (u2h2-u2h1)*(4/dt)-u2h1dot;

func funcu1hdot = (u1h1-4*u1h2+3*u1h)/dt;
func funcu2hdot = (u2h1-4*u2h2+3*u2h)/dt;

func funcu1hddot = (u1h1dot-4*u1h2dot+3*u1hdot)/dt;
func funcu2hddot = (u2h1dot-4*u2h2dot+3*u2hdot)/dt;

savemesh(Th,"Meshinfo.txt");
plot(Th,ps="Meshimage.eps");
```

```
int k=0;
for (real t=Ts;t<(Tf+dt);t+=dt) {
if (t < 0.000000000854700855){
tcurrent=t;
}
else {
tcurrent=0.0;
}

step1;
// time derivative of displacement at time t+(dt/2)
u1h2dot = funcu1h2dot;
u2h2dot = funcu2h2dot;

step2;
// time derivatives of displacement at time t+dt
u1hdot = funcu1hdot;
u1hddot = funcu1hddot;
u2hdot = funcu2hdot;
u2hddot = funcu2hddot;

// save displacements over time of measurement edge
fout<< "␣" << endl;
fout<< "SIMULATION␣TIME␣(seconds)" << "␣␣␣" << t << "␣␣␣␣␣␣␣CPU␣TIME␣(seconds)␣" << clock() <<
     endl;
for (int jj=0;jj<nolines;jj++){
    for (int ii=0; ii<ppline; ii++){
        fout<< xlocations[ii] << "␣" << ylocations[jj] << "␣␣␣␣" << u2h(xlocations[ii],ylocations[
             jj]) << endl;
    }
}

// new time step values
u1h0 = u1h1;
u1h1 = u1h;
u1h1dot = u1hdot;
u1h1ddot = u1hddot;

u2h0 = u2h1;
u2h1 = u2h;
u2h1dot = u2hdot;
u2h1ddot = u2hddot;

if (k%10 == 0) {savevtk("Wave"+k+".vtk", Th, [u1h,u2h]);}
k++;
}
// save displacements along lines that are parallel to propagation direction
int nolinesprop = 100;
ppline = 440;
real[int] xlocationsprop(nolinesprop);
real[int] ylocationsprop(ppline);
for (int ii=0; ii<nolinesprop; ii++){
    xlocationsprop[ii] = ii*0.000000433861641/(nolinesprop-1);
    }
for (int jj=0; jj<ppline; jj++){
```

```
    ylocationsprop[jj] = jj*0.000000433861641/(ppline-1);
    }
fout<< " " << endl;
fout<< "DISPLACEMENTS ALONG PROPAGATION LINES FOR VERY LAST TIMESTEP (x-location , y-location , y-
    displacement)" << endl;
for (int ii=0; ii<nolinesprop; ii++){
    for (int jj=0;jj<ppline;jj++){
        fout<< xlocationsprop[ii] << " " << ylocationsprop[jj] << "    " << u2h(xlocationsprop[ii
            ],ylocationsprop[jj]) << endl;
    }
}
```

REFERENCES

[1] Bathe, K. J., "Conserving energy and momentum in nonlinear dynamics: A simple implicit time integration scheme," *Computers and Structures*, vol. 85, pp. 437–445, 2007.

[2] Bathe, K. J. and Noh, G., "Insight into an implicit time integration scheme for structural dynamics," *Computers and Structures*, vol. 98-99, pp. 1–6, 2012.

[3] Cantrell, J. H. and Zhang, X.-G., "Nonlinear acoustic response from precipitate-matrix misfit in a dislocation network," Journal of Applied Physics, Volume 84, Number 10, November 1998.

[4] Christensen, R., *Mechanics of Composite Materials*. Livermore, California: Lawrence Livermore Laboratory, University of California, 1979.

[5] Cohen-Or, D. and Kaufman, A., "Fundamentals of surface voxelization," *Graphical Models and Image Processing*, vol. 57, pp. 453–461, 1995.

[6] Doerr, C., "Evaluation of sensitization in aisi 304 and aisi 304l stainless steel with nonlinear ultrasonic rayleigh wave measurements," Master's thesis, Georgia Institute of Technology, Atlanta, GA, USA, 8 2016.

[7] Gelfand, I. and Fomin, S., *Calculus of Variations*. Mineola, New York: Dover Publications, 2000.

[8] Herrmann, J., "Generation and detection of higher harmonics in rayleigh waves using laser ultrasoundl," Master's thesis, Georgia Institute of Technology, Atlanta, GA, USA, 12 2005.

[9] Jolliffe, I., *Principal Component Analysis*. New York: Springer Science + Business Media, LLC, 1986.

[10] Kim, J.-Y., "Models for wave propagation in two-dimensional random composites: A comparative study," *Acoustical Society of America*, vol. 127, pp. 2201–2209, 2010.

[11] Kolsky, H., *Stress Waves in Solids*. Mineola, New York: Dover Phoenix Editions, 1963.

[12] Lehtinen, J., "Time-domain numerical solution of the wave equation," 2003.

[13] Matlack, K., *Nonlinear Ultrasound for Radiation Damage Detection*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, USA, 5 2014.

[14] Niezgoda, S. R., "Delineation of the space of 2-point correlations in a composite material system," *Acta Materialia*, vol. 56, pp. 5285–5292, 2008.

[15] Niezgoda, S. R., "Optimized structure based representative volume element sets reflecting the ensemble-averaged 2-point statistics," *Acta Materialia*, vol. 58, pp. 4432–4445, 2010.

[16] Quintanilla, J. and Torquato, S., "Microstructure functions for a model of statistically inhomogeneous random media," *Physical Review E*, vol. 55, pp. 1558–1565, 1997.

[17] Reddy, J., *An Introduction to the Finite Element Method*. New Delhi, India: Tata McGraw-Hill, 2006.

[18] Rose, J. L., *Ultrasonic Waves in Solid Media*. Cambridge, United Kingdom: Cambridge University Press, 1999.

[19] Scott, K., "Development of nonlinear ultrasonic techniques to assess the microstructural damage of 0.1% and 1% fe-cu steel," Proceedings in Quantitative Nondestructive Evaluation, July 2016.

[20] Thiele, S., "Air-coupled detections of rayleigh surface waves to assess material nonlinearity due to precipitation in alloy steel," Master's thesis, Georgia Institute of Technology, Atlanta, GA, USA, 8 2013.

[21] Vincent, E., "Precipitation of the fecu system: A critical review of atomic kinetic monte carlo simulations," *Journal of Nuclear Materials*, vol. 373, pp. 387–401, 2008.